Université de Toulouse

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :
Institut National Polytechnique de Toulouse (INP Toulouse)

Discipline ou spécialité :

Systèmes automatiques

Présentée et soutenue par :
Jun XIONG

le : jeudi 12 septembre 2013

Titre :

Set-membership state estimation and application on fault detection

---

Ecole doctorale :
Systèmes (EDSYS)

Unité de recherche :
LAAS-CNRS

Directeur(s) de Thèse :
Carine Jauberthie, Directrice de thèse
Louise Travé-Massuyès, Co-Directrice de thèse

Rapporteurs :

Luc JAULIN
Tarek RAÏSSI

Membre(s) du jury :
Patrick DANÈS, Professeur d'Université (UPS, Toulouse), Président du Jury
Luc JAULIN, Professeur d'Université, (ENSTA, Bretagne), Rapporteur
Tarek RAÏSSI, Maître de Conférences (CNAM, Paris), Rapporteur
Vicenç PUIG, Professeur d'Université (UPC, Spain), Examinateur
Carine Jauberthie, Maître de Conférences (UPS, Toulouse), Examinateur
Louise TRAVÉ-MASSUYÈS, Directeur de Recherche (LAAS-CNRS, Toulouse), Examinateur

" ***Étudier sans réfléchir est vain,***
mais réfléchir sans apprendre est dangereux. "

- **Confucius**, 551 av. J.-C.

Remerciements

Cette étude a été réalisée au sein de l'équipe **DI**agnostic, **S**upervision et **CO**nduite (DISCO), dirigée par madame Louise Travé-Massuyès directeur de recherche au CNRS. Je tiens a la remercier chaleureusement de m'avoir proposé et de superviser ce sujet, pour son accueil et les moyens donnés pour réaliser ce travail dans les bonnes conditions. Madame Carine Jauberthie a également accepté la direction scientifique de cette thèse, et ce malgré ses nombreuses responsabilités. Je ne sais comment les remercier pour leur disponibilité, leur patience, leurs conseils et tout ce qu'elles ont pu m'apprendre au cours de ces trois années.

Je remercie vivement M. le Professeur Danès qui m'a fait l'honneur de présider le jury. J'ai été également sensible à l'accueil favorable qu'ont réservé messieurs Jaulin et Raïssi à mon mémoire et je tiens à leur exprimer ma profonde gratitude. Je remercie très vivement Professeur Puig et Madame. Le Gall, d'avoir accepté d'être membre du jury.

Si une thèse ne peut se réaliser hors d'un environnement scientifique, le cadre et les relations humaines sont tout aussi indispensables. A ce titre, j'exprime toute ma reconnaissance aux membres de l'équipe DISCO pour leur soutien qui n'a jamais fait défaut. Merci particulièrement à Madame Christèle Mouclier, et madame Sylvie Riethmuller de INPT pour la résolution de tous les soucis administratifs ; merci à Pauline, Elodie, Renaud, Yannick... (liste non exhaustive) pour leurs accompagnement à la cantine, au gym et à la table de café, leur amitié ; un remerciement est également adressé à ma "génération" de thésards : Nuno, Saurabh, Lyamine, Zukun... pour leur soutien sans condition, pour tous les partages entre nous. Je tiens à remercier toutes les personnes du service du LAAS pour leur accueil et leur sympathie. Je voudrais aussi remercier tous mes amis. Témoins de tant de doutes, sans vous ce mémoire n'aurait peut être jamais vu le jour.

En dernier lieu, je ne sais comment exprimer toute ma gratitude à ma femme, qui a également réussi une thèse pendant ces trois ans ; mes remerciements envers mes parents, pour leurs encouragements et leur soutien sans condition. Plus particulièrement, je dédie cette thèse à ma fille, ma petite Leya, qui est née au cours de ces trois ans, pour son énorme gentillesse de faire sa nuit depuis l'âge de 3 mois, et de me libérant toute les nuits quand je devais travailler...

On dit souvent entre les doctorants, que réussir une thèse, c'est comme élever un enfant : il y des moments difficiles, il y des moment sceptiques, il y a également des moments réjouissants. L'essentiel est de tenir jusqu'au bout, de réfléchir et d'apprendre continûment. Je suis très heureux d'avoir eu deux "enfants" parallèlement et je suis fier des deux. J'ai gardé des moments inoubliable.

Abstract

Modeling dynamic systems requires to account for uncertainties arising from noises impacting the measures and/or ! the dynamics, from lack of knowledge about disturbances, and also from uncertainties on parameter values (tolerance specifications, wear processes). Some of these uncertainties, like measurement noises, can be properly modelled in statistical terms but others are better characterized by bounds, without any additional feature.

In this thesis, motivated by the above considerations, we consider the problem of integrating both statistical ans bounded uncertainties for discrete time linear systems.

Building on the Interval Kalman Filter (IKF) developed by [Chen 1997], we propose significant improvements based on recent techniques of constraint propagation and set inversion which, unlike the IKF algorithm, allow us to obtain guaranteed results while controlling the pessimism of interval analysis. The improved filter is named iIKF. The iIKF filter has the same recursive structure as the classical Kalman filter and delivers an enclosure of all the possible optimal estimates and the covariance matrices. The previous IKF algorithm avoids the interval matrix inversion problem and consequently looses possible solutions. For the iIKF, we propose an original guaranteed method for the interval matrix inversion problem that couples the SIVIA (Set Inversion via Interval Analysis) algorithm and a set of constraint propagation problems. In addition, several mechanisms based on constraint propagation are implemented to limit the overestimation effect of interval propagation within the filter recursive structure.

A fault detection algorithm based on the iIKF is proposed. It implements a semi-closed loop strategy which stops feeding the filter with observation corrupted by the fault as soon as it is detected. Through various examples, the advantages of the iIKF filter are presented and the effectiveness of the of the fault detection algorithm is demonstrated.

Resumé

La modélisation des systèmes dynamiques requiert la prise en compte d'incertitudes liées à l'existence inévitable de bruits (bruits de mesure, bruits sur la dynamique), à la méconnaissance de certains phénomènes perturbateurs mais également aux incertitudes sur la valeur des paramètres (spécification de tolérances, phénomène de vieillissement). Alors que certaines de ces incertitudes se prêtent bien à une modélisation de type statistique comme par exemple ! les bruits de mesure, d'autres se caractérisent mieux pa ! r des bornes, sans autre attribut.

Dans ce travail de thèse, motivés par les observations ci-dessus, nous traitons le problème de l'intégration d'incertitudes statistiques et à erreurs bornées pour les systèmes linéaires à temps discret.

Partant du filtre de Kalman Intervalle (noté IKF) développé dans [Chen 1997], nous proposons des améliorations significatives basées sur des techniques récentes de propagation de contraintes et d'inversion ensembliste qui, contrairement aux mécanismes mis en jeu par l'IKF, permettent d'obtenir un résultat garanti tout en contrôlant le pessimisme de l'analyse par intervalles. Cet algorithme est noté iIKF. Le filtre iIKF a la même structure récursive que le filtre de Kalman classique et délivre un encadrement de tous les estimés optimaux et des matrices de covariance possibles. L'algorithme IKF précédent évite quant à lui le problème de l'inversion des matrices intervalles, ce qui lui vaut de perdre des solutions possibles. Pour l'iIKF, nous proposons une méthode originale garantie pour l'inversion des matrices intervalle qui couple l'algorithme SIVIA (Set Inversion via Interval Analysis) et un ensemble de problèmes de propagation de contraintes. Par ailleurs, plusieurs mécanismes basés sur la propagation de contraintes sont également mis en œuvre pour limiter l'effet de surestimation due à la propagation d'intervalles dans la structure récursive du filtre.

Un algorithme de détection de défauts basé sur iIKF est proposé en mettant en œuvre une stratégie de boucle semi-fermée qui permet de ne pas réalimenter le filtre avec des mesures corrompues par le défaut dès que celui-ci est détecté. A travers différents exemples, les avantages du filtre iIKF sont exposés et l'efficacité de l'algorithme de détection de défauts est démontré.

" ***Sic infit...*** "

Contents

List of Figures

List of Tables

Introduction

Set-membership (SM) methods have been the focus of a growing interest and they have been applied to many tasks ([Gelso 2008, Jaulin 2001a, Kieffer 1999]). The litterature on this topic shows interesting progress in the last years. SM estimation can be based on interval analysis that was introduced by [Moore 1966] and several algorithms have been proposed (for more details, see [Jaulin 2001d, Ribot 2007, Kieffer 1999]). Other approaches dedicated to linear models include ellipsoid shaped methods ([Milanese 2004, Lesecq 2003]), parallelotope and zonotope based methods [Ingimundarson 2009].

One of the main advantages of the SM estimation approach is that it provides a guaranteed solution in contrast to stochastic estimation approaches. However, it does not give any precision about the belief degree.

On the other hand, the SM approach is often criticized for the overestimation of the results. However, one should note that in a similar way, the stochastic approach may estimate system states with a wide confidence range, which may turn as difficult the interpretation of the results.

As a matter of fact, both techniques have specific advantages and they can interact synergically. A stochastic method can provide means for analyzing the properties of an SM estimator, and conversely an SM technique can provide the initial entry to a system without assuming distribution law in advance. They are hence more complementary than competitive.

Most importantly, in an estimation framework, the experimental conditions about noise and disturbances are usually properly modeled through appropriate assumptions about probability distributions. However, other sources of uncertainty are not well-suited to the stochastic approach and are better modeled as bounded uncertainty. This is the case of parameter uncertainties that generally arise from design tolerances and from aging. In such cases, combining stochastic and bounded uncertainties may be an appropriate solution.

Motivated by the above observations, we consider the filtering problem for discrete time linear models with bounded uncertainties on parameters and Gaussian measurement noise. In [Chen 1997], the classical Kalman filter [Kalman 1960] has been extended to interval linear models. We consider this work and propose several operations that improve the algorithm. In particular, the approach proposed in [Chen 1997] does not provide guaranteed results because it avoids especially interval matrix inversion by using operations described below. Our contribution consists in proposing an alternative approach to solve the interval matrix inversion problem without loss of solutions while controlling the inherent pessimism of interval calculus. Several technical operations are proposed to limit

the overestimation effects due to interval propagation within the interval Kalman filter recursive structure. In particular the gain of the filter is obtained by a calculus based on the set inversion algorithm SIVIA (Set Inversion via Interval analysis) [Jaulin 2001d] which is combined with constraint propagation techniques.

The main contributions of the thesis consist of the following points:

- an alternative approach to solve the interval matrix inversion problem without loss of solution while controlling the inherent pessimism of interval calculus;
- a set-membership state estimation approach for linear systems based on the Kalman filter, which copes with parameter bounded uncertainty and statistical noise;
- a combination of techniques has been implemented to limit the overestimation effects propagating within the interval Kalman filter (IKF) recursive structure;
- a new way to evaluate the filter gain based on the set inversion algorithm SIVIA; the results have shown that the improved interval Kalman filter includes a family of optimal estimates derived from bounded error and the result is better than existing approaches;
- an adaptive threshold which defines the healthy model of the system is established by combining the set-membership analysis and statistical behaviour of the noise based on the improved Interval Kalman filter; it is shown that this method is very useful for detecting additive sensor faults on measurements.

The work is organized as follows: Chapter 2 presents appropriate tools including the interval analysis and related works for set-membership approach, an important improvement on interval matrix inverse is proposed; Chapter 3 explains the basic definition of uncertain stochastic system, and our state estimation approach based on the existing Interval Kalman Filter; the next Chapter uses the improved IKF to generate an adaptive threshold for fault detection. Case studies are proposed in order to explain the implementation of the proposed algorithm, from academical examples to the aerospace application simulation.

Chapter I
State of the art

CHAPTER 1

State of the art: fundamental motivation

Contents

1.1 Introduction

Any system has input(s), output(s), and a given system structure serving designed functions. Generally, we can easily access the system input and output, hence their values

are known; however on the other hand, the system states are not always accessible or known in advance, or they are just partially known. Moreover, the parameters could vary during the running process according to the circumstance's change, such as temperature or altitude change. These derivations make it difficult to obtain the real value of system parameters or states.

Meanwhile, the system state is important to many tasks. For instance, state feedback can be used in a control loop in order to stabilize the system or tweak the system behaviour (speed, accuracy). It can also contribute to supervising the system, in tasks like fault detection or system prognosis. Generally saying, the more informations the controller or supervisor has about the system state, the better (more accurate, faster...) the controller/supervisor can react.

In many practical cases, the system physical states are not easy to obtain by direct observation, due to the structure complexity, or the limitation of sensors. Take an Inertial Navigation System (INS) for example. The initial position of the system is known in advance, the system attitude and velocity are measurable, but the actual geographical position of the system cannot be measured directly by the INS itself. In order to get the exact position of the system at any given time, exterior observation is needed. It is possible to use a Global Navigation System (like GPS, GLONASS or COMPASS) receiver to obtain the precise position, which is called "hardware" observer. However, the extra hardware adds up the cost, weight, energy resource consumption and complexity of wiring or structural design. For some applications, these extra demands are hard to satisfy.

It is then interesting to reconstruct the system state from available information using only "software" methods. This technique is called "software sensor", or state estimation. It is designed to provide the internal state of the real system, by using the measurements of the physical input and output, along with the knowledge of an abstracted mathematical model which represents the physical system.

The idea of state observer was firstly introduced by Kalman on linear systems in the stochastic context [Kalman 1960], but early work [Jeffreys 1946] can be tracked back to the 40's. Since then, many new approaches have been developed and implemented, for linear systems as well as non-linear systems. Important aspects of the estimation theories include the robustness, efficiency and flexibility of the estimators.

Our work focuses on state estimation, when the system is affected by both uncertainties and noises. It is obvious that such a context represents a big challenge for establishing a suitable state observer, which is desired to provide the correct range under types of uncertainties existing in the system structure and sensors. A state estimation well adapted to multi-type uncertainties can be used in system supervision, especially for fault detection. Conventional approaches indeed suffer from the inability of the system model to properly consider uncertainties.

In this chapter the basic concept about state estimation theory is presented. Different approaches of estimation are introduced in section 1.3, from linear system to non-linear systems. Different types of uncertainty models are presented in section 1.2. In Section

1.4 the set-membership state estimation problem is formalized and its essential elements are described. Various approaches are reviewed both for linear systems and non-linear systems. In Section 1.5 the application of state estimation to fault detection is presented. The last Section 1.6 introduced the objective of this work and our main contribution.

1.2 Modelling and uncertainties

Modelling, in the general sense, aims to provide a mathematical representation with suitable algebraic characteristics. All models are simplified reflections of reality. The model structure should describe the system behaviour of interest.

Building models is fundamental to any research. There exists no complete and perfect representation of reality, the objective of modeling is to find the best model for a given task, and construct a formal system whose theoretical properties are not contrary to the behaviours found in reality. Keep in mind that *no model can represent the real system*. There are several basic categories of mathematical models, such as linear or non-linear system models, static or dynamic models, discrete or continuous time models, and deterministic or stochastic model. The choice is made based on the knowledge of the analysed real process and controller/supervisor's objective, and most importantly, according to how much *a priori* information is available. Usually it is preferable to use as much *a priori* information as possible to make the model more accurate.

Since a model only refers to some aspects of the studied phenomenon in question, different models of the same phenomenon may be essentially different, due to different requirements of the model's and users, or to conceptual differences among the modellers. For example, the continuous time systems have both temporal and frequency representations, they represent both the model from the continuous system behaviour point of view, but in different applications they have different advantages or disadvantages.

The system modeling, especially for confirming system parameters or approximating a non deterministic variable, is done by extracting information from empirical data. This information is called subjective information. This kind of information is modelled by experimental experience, or expert knowledge. For instance, the Bayesian statistics specifies a prior probability distribution which can be subjective and updates the information based on new empirical data. A model should be evaluated according to wheater it fits empirical data, wheater it is appropriate and consistent with the prior assumptions.

One of the key points of this work is how to model the uncertainties in a dynamic system. The stochastic framework is one way to represent uncertainties which have *a priori* statistical characteristics. But there are also uncertainties which do not necessarily have an explicit distribution laws or simply the law is not known. A proper model of uncertainty representation for non statistical uncertainty is needed. In next section, we present the origins of uncertainties and compare several forms of uncertainty approximation that we may use in our work.

1.2.1 Origins of uncertainties

It is important to notice that uncertainty in the system is considered and analysed separately from other "model deviations", such as faults, though there are certain similarities among them. They are all deviations from the nominal system configuration, nominal specification or nominal performance, but contrary to faults, uncertainties are acceptable deviations within the tolerances of the process. A complete presentation on uncertainty and imprecision for the systems can be found in [Bloch 1996].

Where do the uncertainties come from? Take an electronic system for example: a typical tension-resistance-current mechanism. The system can be modelled by $V = IR$. The ideal value of the resistance is R_0 which we consider as the nominal value. The voltage at the resistance bounds is V and the current getting through the resistance is I. According to the Ohm law, V and I are proportional.

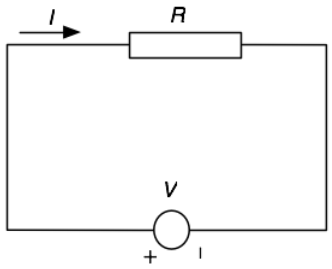

Figure 1.1: Ohm law system

The tension-resistance-current relation is valid with several assumptions. Firstly, the measurement instruments are assumed to be perfect as the value of I and V are measured without any errors or noises, noted by I_m and V_m. Secondly, the model $V = R_0 I$ is deterministic when the current I_m corresponds to the perfectly measured tension V_m with fixed resistance R_0.

However in practice, it is nearly impossible to have a strictly ideal component. The manufacturer always gives a resistance value with certain precision σ where $\sigma = 5\%, 10\%....$ it shows the value of R is knowingly included in an interval $[(1-\sigma)R_0, (1+\sigma)R_0]$.

Secondly the true value of the resistance is also affected by the environment. For instance the temperature changes may change the resistivity. It is generally difficult to know precisely the changing coefficient. In conclusion, the value R of the resistance also refers to a bounded precision and deviates from its nominal value R_0 because of various perturbations.

This is why an uncertain model is introduced in place of a deterministe model $V = R_0 I$ to properly interpret the set of possible real values. Not only the resistance is a component with uncertainty, but the current I and the tension V are also measured with limited precision. This is in addition to the probabilistic noise at the sensors. A simple uncertain

model of measured current I_m can be written as:

$$I_m = (1 + \Delta)I + \theta,$$

in which the parameter Δ represents the relative imprecision of current sensors because of the non-linearity due to the saturation, and the parameter θ represents a random noise.

This example shows that the model for a given system is not unique, different approximations can be made according to different modellings. It is important to involve uncertainties in the system model so that we do not lose solutions. There exists two major categories of model uncertainties: structural uncertainty and parametric uncertainty. Different mathematical models can be used to approximate different types of uncertainty.

1.2.2 Uncertainty modeling

Two main approaches are used to approximate uncertainties: the stochastic approach and the Set-membership approach.

Stochastic The stochastic approach consists in approximating a random vector X of $\mathbb{R}^n$ by a density function or distribution law f, a vectorial function which verifies the following property:

$$f : \mathbb{R}^n \to \mathbb{R} \mid \int_{\mathbb{R}^n} f(x)dx = 1.$$

The stochastic model gives two types of information. On one side it provides the support of the uncertain variable, which is all possible values for this variable; on the other hand, it also generates the value distribution on this support. Classically, the uncertainty is represented in this form as an additive term to approximate the noises. Most distribution laws have unbounded support, such as gaussian distribution for example.

The stochastic approach has its limit when the distribution law is difficult to know. One of the major criticisms of this approach is that the requisite distributions are often not available. To solve the problem, one can use more empirical data to "train" the model, if the statistical characteristics exist among the data [Ferson 2002]. However, it is not always possible to find a proper distribution information for describing the uncertainty's behaviour, or to have enough data and time to obtain a precise stochastic model. There are also cases where the statistical information is partially known, or the known distribution is not precise. Abundant researches can be found in this domain varyingly naming "*imprecise probability*" or "*credal sets*" or "*p-boxes*". In these approaches the distribution parameters are represented by set values, such as an interval variance or set mean value [Wang 2008a, Walley 1991, Hall 2004, Kreinovich 2004, Utkin 2009, Cuzzolin 2010, Alvarez 2009, Cozman 1997, Abellán 2005], a unified theory of these approaches can be found in [Walley 2000]. Their applications on estimation cover also very large area ([Kai 2010, Klumpp 2009, Batarseh 2008, Yager 1999, Noack 2009, Hable 2010, Rico 2010, Noack 2008]).

Another complex scenario is when multi sensors are used to estimate the same state, each sensor has its own distribution information, the fusion of multi sensors' measurement generates the problem of confidence degree on the estimation result [Zhu 2006, Shi 2007, Kosanam 2004].

The above approaches are still based on statistical assumptions. The approximation is based on the statistical distribution. In some other cases where the statistical distribution does not exists, set-membership approach is an alternative.

Set-membership The set-membership approach approximates the uncertainty by giving a set of values with equal "weight". In the set they are all "possible" candidate values. This approach is poorer on providing information compared to stochastic approaches, as it only retains the information on the boundaries of the variable support, and not much about the distribution between the boundaries. However, modeling with bounded variables is often a direct translation of the available knowledge. When no distribution is known, the set-membership approach is more appropriate than the stochastic approach. Moreover, when system becomes more and more complex, the stochastic framework is difficult to tweak and implement, but recent development make the set-membership approach applicable in these cases.

There are different bounded error approximations, for instance, without being exhaustive, one may use different geometrical regions involving an appropriate machinery such as the *ellipsoids* [Calafiore 2004, Poignet 2003b], the *parallelotopes* [Hero 1997], the *polytopes* [Ziegler 1995] , the *zonotopes* (a particular polytope) [Bourgain 1989] or the *orthotopes* [Milanese 1982], one can also use a simple *interval* approximation. The combination of different geometrical regions is also possible [Kreinovich 2009].

The boundary determination which interprets the parametric uncertainty of the model is called the characterization step. This step determines the set of acceptable values of the parameters with which the model can better explain the data, under the chosen criterion and the available observations [Milanese 1991]. Depending on the strategy and the given model, the domain can be obtained precisely or approximated with adequate methods.

The paving method can be used to get an approximation of the domain for any kind of complexity. In [Adrot 2002b], the authors concentrate on analyzing the bounded variables in a linear model in order to find the influence of the uncertainty on parameters. The paving method is used to determine the interval of the uncertainty under the criterion of model precision.

Our work is based on interval analysis [Moore 1966]. With this approximation, an uncertain model defines not only one behavior reference but a set of possible or acceptable behaviors for the system, at the same time the form of uncertainty zone is convex and simple enough so that the result can be easily used.

Proper modeling of uncertainty is the foundation of further works, such as actuator control [Canale 2009], system lifetime prediction [Maksarov 1998], and of causes state estimation. In the next section, as a preliminary part, we introduce the basic principle of

state estimation following by several approaches for linear and non-linear systems.

1.3 State estimation

Consider a system model described as:

$$\begin{cases} \dot{x} = f(x, e, \theta, u), \\ y = g(x, \varepsilon, \gamma, u). \end{cases} \tag{1.1}$$

where y represents the output vector of the system, u the input vector, x the state variables which are not entirely measurable, θ and γ the system parameters, e and ε the modeling errors and/or noise terms affecting the process. θ and γ are considered as constant whereas the other variables are function of time t, f and g are non-linear functions. The objective of estimation is to get desired states from measurements of y, u by a suitable procedure without using additional physical sensors which can be expensive.

A typical state observer structure can be presented as following, as x is assigned the system state, we use $\hat{x}$ to represent the state estimation generated from the observer.

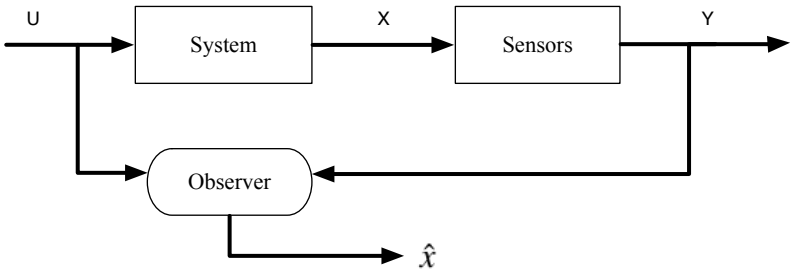

Figure 1.2: State observer structure

It is shown in the figure 1.2 that the state estimation acts as the soft sensor of the system dynamics by using the available input and output values, aiming to correct the difference between the reality and the model. The estimation problem can be transformed into an equivalent optimization problem, by maximizing or minimizing certain cost function or condition. The estimate can approximate the real state as executing time increases in real-time applications or as the empirical data becomes more about in off-line reconstruction. That is to say that an estimator can "learn" how to approach the real state.

The state estimation problem can be solved better when the number of measurements increases, but the size of the optimization process increases at the same time in some methods. To limit the memory space and computational time, one may use a sliding horizon approximation to give a bounded size of optimization with approximate estimator to speed up the algorithm [Drakunov 2010, Alessandri 2000, Puig 2003] especially for non-linear system. In a way, there is a choice between accuracy and speediness.

For conventional cases which are not in the set-membership context, there are different observer methods covering from linear systems to non-linear systems. In the following

section we present several classical approaches for each system type. They are the foundation of further work in the set-membership context. As there may exist noises in the system structure and sensors, the estimator consists in extracting useful information from the noise pollution, the estimator is also called "filter". The methods proposed in the next sections can be used to estimate the state with or without noises and uncertainties.

1.3.1 Linear model case

Firstly the general theory of observer is proposed by Luenberger for a deterministic linear system without stochastic modeling on noises, it is not suited to all systems according to [Luenberger 1964, Luenberger 1971]. Observers are usually developed based on the modeling of linear system with Gaussian noises, which can be given by the form:

$$\begin{cases} x_{k+1} = Ax_k + Bu_k + w_k, \\ y_k = Cx_k + Du_k + v_k, k = 0, 1, 2, ... \end{cases} \tag{1.2}$$

where x_k, y_k and u_k are defined in Equations (1.1), $x_k \in \mathbb{R}^n$, $y_k \in \mathbb{R}^m$, $u_k \in \mathbb{R}^p$. If the system is observable, the output of the system y_k can be used to derive the state estimate. The matrices $A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times p}, C \in \mathbb{R}^{m\times n}$ and $D \in \mathbb{R}^{m\times p}$ are constant matrices; $A, B \subset \theta$, $C, D \subset \gamma$ are parameter sets referring to the Equation (1.1); $\{w_k\}$ and $\{v_k\}$ are independent Gaussian white noise sequences, with respective covariance matrices Q and R. Q and R are definite positive by definition:

$$\begin{aligned} E\{w_k, w_l\} = Q\delta_{kl}, E\{v_k, v_l\} = R\delta_{kl}, \\ E\{w_k, v_l\} = E\{w_k, x_0\} = E\{v_k, x_0\} = 0, \\ \forall k, l = 0, 1, 2, ... \end{aligned}$$

where δ_{kl} is the Kronecker symbol.

As the state observer is generally computer-implemented, the discrete-time representation has more advantage for actual implementation and simulation. The process and equations are similar for the continuous-time case.

1.3.1.1 Kalman filter

One of the well-known filters for linear discrete system is the Kalman filter, formally introduced in [Kalman 1960]. It provides state and noise variance estimate of the linear dynamic system from a series of noisy measurements. The main assumption of the Kalman filter is that the underlying system is a linear dynamical system and that all error terms and measurements have a Gaussian distribution (often a multivariate Gaussian distribution).

There are several ways to deduce Kalman equations: one can use mathematical curve-fitting function of data points from a least-squares approximation [Welch 2001]; one can

also use more probabilistic methods such as the Likelihood function to maximize the conditional probability of state estimate from measurement incomes [Masreliez 1977]. The latter shows the connexion between the Kalman filter and the recursive Bayesian estimation. The Kalman filter is a recursive estimator, meaning that only the estimated state from the previous time step and the current measurement are needed to compute the estimate of the current state.

Te following notations are used:

- $\hat{x}_{k+1|k} \in \mathbb{R}^n$ the *a priori* state estimate vector at time $k+1$ given state estimate at time k,
- $\hat{x}_{k|k} \in \mathbb{R}^n$ the *a posteriori* state estimate vector at time k given observations at time k,
- $P_{k+1|k} \in \mathbb{R}^{n\times n}$ the *a priori* error covariance matrix,
- $P_{k|k} \in \mathbb{R}^{n\times n}$ the *a posteriori* error covariance matrix.

$P_{.|.}$ is an essential indicator that defines the estimated accuracy of the state estimate. It is defined as:

$$P_{k|k} = E\left((x_k - \hat{x}_{k|k})(x_k - \hat{x}_{k|k})^T\right), \tag{1.3}$$

$$P_{k+1|k} = E\left((x_{k+1} - \hat{x}_{k+1|k})(x_{k+1} - \hat{x}_{k+1|k})^T\right). \tag{1.4}$$

It is known that the Kalman filter algorithm contains two steps for each iteration: a prediction phase and a correction phase, and the whole recursive algorithm begins with an initialization phase. The algorithm is presented as following:

Initialisation:

$$\begin{aligned} \hat{x}_{0|0} &= E(x_0), \\ P_{0|0} &= Cov\{\mathbf{x}_0\}. \end{aligned}$$

prediction:

$$\begin{aligned} \hat{x}_{k+1|k} &= A\hat{x}_{k|k} + Bu_k, \\ \hat{P}_{k+1|k} &= A\hat{P}_{k|k}A^T + Q, \\ k &= 0, 1, 2, ... \end{aligned}$$

correction:

$$\begin{aligned} K_{k+1} &= \hat{P}_{k+1|k}C^T(C\hat{P}_{k+1|k}C^T + R)^{-1}, \\ \hat{P}_{k+1|k+1} &= (I_n - K_{k+1}C)\hat{P}_{k+1|k}, \\ \hat{x}_{k+1|k+1} &= \hat{x}_{k+1|k} + K_{k+1}(y_{k+1} - Du_{k+1} - C\hat{x}_{k+1|k}), \\ k &= 0, 1, 2, ... \end{aligned}$$

A necessary condition for the Kalman Filter to work correctly is that the system for which the states are to be estimated, is observable. Therefore, you should check for observability before applying the Kalman Filter. Usually it is necessary to fine-tune the Kalman Filter when it is connected to the real system. The process noise auto-covariance Q and/or the measurement noise auto-covariance R are commonly used for the tuning.

An important advantage of Kalman filter is that it does not need to memorize the measurements history and it gives a better result to state estimation because it is easier and more intuitive to tune the filter in terms of process and measurement noise variances; its discrete time version can be easily implemented in system simulation or process reconstruction by using only MATLAB, both for academical or practical purpose [Grewal 2001]. It advances other approaches in real time application [Chui 1987] so that it becomes nearly a synonym of the linear state estimation.It has lots of variations and has been implemented in many applications, such as automatic control [Lou 2003], tracking and navigation [Hoehmann 2007], signal processing [Macalak 2010], finance analysis [Wells 1995], even in computer language [Laaraiedh 2009]. Extensions and generalizations to the method have also been developed, such as the Extended Kalman filter and the Unscented Kalman filter which work on non-linear systems; details can be found in the next section on non-linear system.

The Kalman filter also belongs to a larger catalogue naming "minimum variance filters" or "H_2 estimator". Similar filters such as Wiener filter require an exact model of signal generating process and full information about a statistical nature of noise sources.

1.3.1.2 Recursive least squares filter

Another frequently used prediction-correction based filter-estimator for linear system is called the Recursive least squares filter (RLS) [Zhu 1999]. Similar to the Kalman gain coefficient in the Kalman filter which can minimize the error covariance, this method also uses a filter coefficient that can minimize a weighted linear least squares cost function related to input signal. However in this algorithm, the input measurement is considered determinant. The idea behind RLS filter is to minimize a cost function by appropriately selecting the filter coefficients updating the filter as new data arrives. The algorithm structure is as following:

Initialization:

$$\hat{x}_0 = x_0, P_0 = \alpha I, \lambda. \tag{1.5}$$

Recursion:

$$\begin{aligned} L_{k+1} &= P_k A^T C^T (\lambda I_n + CAP_k A^T C^T)^{-1} \\ \hat{x}_{k+1} &= A[\hat{x}_k + L_{k+1}(y_{k+1} - CA\hat{x}_k - Bu_k)] \\ P_{k+1} &= \lambda^{-1}(I_n - L_{k+1}C^I A)P_k A^T, \\ k &= 1, 2, ... \end{aligned} \tag{1.6}$$

where $\alpha > 0$ is the parameter defined by the user, and λ is the forget factor, $0 < \lambda \leqslant 1$. By tweaking the forget factor, the estimator has a choice between depending more on current income measurement or more on historical estimates.

The RLS filter is useful when noise covariances are difficult to obtain, in which case the Kalman filter is less efficient.

1.3.1.3 The observers

Many sophisticated analytical procedures for control design are based on the assumption that the full state vector is available for measurement. In many systems of practical importance, however, the entire state vector is not available for measurement. The approach that constructs an approximation to the full state vector on the basis of available measurements is called observer. Any of the earlier static control procedures can then be implemented using this approximate state in place of the actual state. It is shown that the state (or an approximation to it) can be conveniently computed by a device known as an observer [Luenberger 1971].

A more general situation can be expected, in which a fully-observable, linear, time-invariant system has state x_k of dimension n, input u_k, and output y_k of dimension p with state and output equations as Equation (1.2).

The output y_k gives us information about the state of the system, even if somewhat indirectly. In particular, the exact copy of state equation can be augmented by incorporating a term depending on the difference between the observed output and the expected one. We obtain, for some $n \times p$ matrix L, the following structure:

$$\hat{x}_{k+1} = A\hat{x}_k + L(y_k - \hat{y}_k) + Bu_k, \tag{1.7}$$

It is clear that unless $p = n$, the observer gain matrix L is needed in order to obtain a term of the correct dimension. The choice of the observer gain matrix can also affect the dynamic behavior of the state estimate and thus the state error.

Some observers have slightly more complicated structure in that it has two input vectors, u_k and y_k, but a better dynamic response can be expected. This is called Luenberger observer, and many extensions and variations ([Bourij 1999, Chaves 2002, Fairman 1977, Hou 1992, Nandam 1990, Orlowska-Kowalska 1989, Price 1982, Xu 2005, Zeitz 1987]) have been developed based upon it.

1.3.1.4 Other linear methods

Other filters than Kalman filters and the RLS filter exist: there are also the least mean squares filter (LMS) and its variations. In the LMS filter, the filter weights are updated in order to converge to the optimum filter weight, It differs from the RLS filter; the input signals are considered stochastic, and the chosen cost function is the mean square error. As a matter of fact, unlike under the non-linear context, for linear dynamic system, the

Kalman filter is nearly the standard solution. In our work, we suppose that the noise characteristic is known, so that the Kalman filter is more suitable than other methods.

1.3.2 Non-linear model case

In the linear case, if the system is observable, we can find one state which is the optimal estimate, elsewhere there are infinitely many states are optimal estimates; if the system is non-linear, it is possible to have finitely many states which are locally optimal estimates, which adds the difficulty on the estimation.

The modelling of non-linear system with Gaussian noises can be given by the following equations:

$$\begin{cases} x_{k+1} = f(x_k, u_k, \theta) + w_k, \\ y_k = h(x_k, u_k, \gamma) + v_k, k = 0, 1, 2, ... \end{cases} \tag{1.8}$$

where x_k,y_k and u_k are defined in Equations (1.2), f and h are non-linear functions, θ and γ are defined in Equations (1.1).

To solve the estimation problem for a non-linear system, it is possible to transform it into a traditional linear system problem using linearizion of the original equations; one can also use probabilistic approach to approximate the system evolution.

1.3.2.1 Extended Kalman filter

The first filter for non-linear system using linearizion is the extension of classical Kalman filter (EKF). In fact, it is considered as the standard non-linear state estimation for well defined models [Julier 2004].

The EKF uses multivariate Taylor Series expansions in order to linearise the working point. The system equations are assumed to be differentiable functions. A Jacobian matrix of partial derivative is needed to calculate the covariance prediction and estimation. At each iteration the Jacobian matrix is evaluated with present predicted states.

The algorithm is presented as following:

Initialisation:

$$\begin{aligned} \hat{x}_{0|0} &= E(x_0), \\ P_{0|0} &= Cov\{x_0\}. \end{aligned}$$

prediction:

$$\begin{aligned} \hat{x}_{k+1|k} &= f(\hat{x}_{k|k}, \theta, u_k), \\ \hat{P}_{k+1|k} &= F_{k-1}\hat{P}_{k|k}F_{k-1}^T + Q_{k-1}, \\ k &= 1, 2, ... \end{aligned}$$

correction:

$$\begin{aligned} K_{k+1} &= \hat{P}_{k+1|k} H_k^T (H_k \hat{P}_{k+1|k} H_k^T + R_k)^{-1}, \\ \hat{P}_{k+1|k+1} &= (I_n - K_{k+1} H_k) \hat{P}_{k+1|k}, \\ \hat{x}_{k+1|k+1} &= \hat{x}_{k+1|k} + K_{k+1}(y_{k+1} - g(\hat{x}_{k+1|k}, u_{k+1})), \\ k &= 1, 2, ... \end{aligned}$$

state transition and observation matrices:

$$\begin{aligned} F_{k-1} &= \left.\frac{\partial f}{\partial x}\right|_{\hat{x}_{k-1|k-1}, u_{k-1}}, \\ H_k &= \left.\frac{\partial h}{\partial x}\right|_{\hat{x}_{k|k-1}}, \\ k &= 1, 2, ... \end{aligned}$$

The extended Kalman filter (EKF) is probably the most widely used estimation algorithm for non-linear systems [Edelmayer 2007, Blanchard 2007]. However, more than 35 years of experience in the estimation community has shown that is difficult to implement, difficult to tune, and only reliable for systems that are almost linear on the time scale of the updates . In addition, if the initial estimate of the state is wrong, or the modeling is incorrect, the filter may diverge.

EKF has several variations, for example, there is Robust extended Kalman filter which emphasizes the stability over the optimality [Welch 2001, Xue 2007, Kai 2010]; there is also Unscented Kalman filter (UKF) where the probability density is approximated by a deterministic sampling of points representing the underlying distribution as a Gaussian; UKF is supposed to be more robust and accurate than the original EKF [Julier 2004, Tudoroiu 2011]. The ensemble Kalman filter (EnKF) is also an alternative to the EKF [Evensen 2003]: it circumvents the expensive integration of the state error covariance matrix by propagating an ensemble of states from which the required covariance information is obtained at the time of the update. The application of EnKF can be found in [Reichle 2002].

1.3.2.2 Particle filter

Another approach to estimate state under the non-linear context other than the linearisation is using the probability theory and simulation approximation. Let us use $p(.|.)$ a generic conditional distribution, the general discrete time state-space model can be also given by:

$$\begin{cases} x_k \sim p(x_k \mid x_{k-1}; \theta), system \\ z_k \sim p(z_k \mid x_k; \gamma), observation \end{cases} \tag{1.9}$$

where z_k contains the observations at time t_k, whereas $\{x_k\}$ is an unmeasurable stochastic state. θ is a vector containing static parameters, which in some cases can be specified but in

many cases are unknown. Typically, some prior distribution is placed on x_0. The objective here is the estimation of the state process based on measurements from the observation process. For situations where analytical solutions such as EKF are impossible to obtain, stochastic simulation can be applied.

Numerous papers ([Gamerman 1998, Hürzeler 1998]) have been written on construction of algorithms based on Markov chain Monte Carlo (MCMC) dealing with general state-space models. But the MCMC algorithm needs to be restarted at each time point. An alternative to full MCMC at each time point is construction of simulation algorithms for sequential updating of the posterior distributions. Such algorithms have been developed independently in many fields [Doucet 2001a, Crisan 2002, LeGland 2000, Doucet 1998] with different names (bootstrap filter, sequential Monte-Carlo method, particle filter, condensation algorithm...). The name "particle filter" (PF) is mostly adopted by the community.

The main idea behind particle filter is to represent the posterior distributions $p(x_{1:k} \mid z_{1:k})$ through a finite set of samples or particles that can be used to estimate any property of $p(x_{1:k} \mid z_{1:k})$ in an ordinary Monte Carlo estimation framework. When a new observation z_{k+1} arrives, the particles are updated in order to represent the new posterior $p(x_{1:k+1} \mid z_{1:k+1})$.

This method approaches the a posterior by the mean value of all the measurement points, which resemble "particles" with weight. The classical Monte-Carlo method was developed in the 50's, in spite of the computation capacity limit at that time. Since the 90's, as the computer science boosted and the power of modern computer burst rapidly, the Monte-Carlo method has redrawn the attention and has been largely developed. The particle filters became useful in actual application especially when the sequential importance resampling (SIR) was proposed in 1993 [Gordon 1993]. Recent works still focus on finding the balance point of algorithm speed and estimate error control [Faubel 2008, Verma 2003]. A complete review of these works can be found in [Doucet 2001a, Oppenheim 2008].

This is essentially an estimation technique based on simulation. In fact, particle filters can cope with all dynamic Markov systems without any knowledge of noises characteristics or of the time-varying parameters in signal processing system [Doucet 2001b]. It can perform state estimation on wide range of system for fault detection, system identification and control. A detailed overview can be found in [Andrieu 2004]. It is also very suitable for hybrid systems [Funiak 2003, Koutsoukos 2003, Hutter 2003, Hofbaur 2004].

The applications on estimation and fault diagnosis using particle filtering can be found in [Li 2001, Mcilraith 2000, Verma 2004]

1.3.2.3 Other non-linear model cases

Other non-linear filters which are worth mentioning include Volterra filtering which uses the Volterra series expansion for linearizion. It differs from the Taylor series expansion: the Volterra series expansion is a linear combination of non-linear functions of the input signal. It has the memory in the precedent states [Morrison 1991].

There is also Moving horizon estimation (MHE) or Sliding-mode observer and filter [Alessandri 2000]. It is an optimization approach that uses a series of measurements observed over time, containing noise (random variations) and other inaccuracies, and produces estimates of unknown variables or parameters. Unlike deterministic approaches like the Kalman filter, MHE requires an iterative approach that relies on linear programming or non-linear programming solvers to find a solution.

In [Storvik 2002] authors deal with unknown static parameters (somewhat "uncertain"); the solution however concerns the Kalman filter based parameter identification which can help updating the system parameter information. Another interesting work consists in representing the particles in PF by interval [Abdallah 2007]. There are few works under the same context as we are; particle filter has the advantage of coping with various type of non-linear systems. In the perspective of our work, we consider using the particle filter as the foundation of set-membership extension of our approach for non-linear system.

1.4 Set-membership estimation

Bounded error approaches allow the characterization of the set of all values of the state vector that are consistent with measured data, the model structure and the prior known error bounds. Available methods based on set-membership (SM) approaches exist for linear and non linear models. In the set-membership context, the estimation of the physical quantity (state or parameter) encloses the set of solutions or a union of disconnected sub-sets of solutions which are produced by the uncertainty of experimental data and the predefined system error boundaries.

In this work, we consider that the system parameters are uncertain but bounded. They belong to a known-in-advance set of values. For example, a discrete linear system with set valued parameter:

$$x_{k+1} = \boldsymbol{A}x_k + u_k,$$

where $\boldsymbol{A}$ is an interval matrix, the definition of interval matrix is presented in the next chapter along with other tools of the interval analysis. Of course sets can have different geometrical forms other than boxes. In general, the set will be presented in bold type through out this work.

The hypotheses for the system can be presented by the following equations:

$$\begin{cases} x_{k+1} = f(x_k, u_k, p), \\ y_k = g(x_k, u_k, p), \\ x_0 \in \boldsymbol{X}_0, \\ p \in \boldsymbol{P}, \end{cases} \tag{1.10}$$

where f and g are the known non-linear functions; $x_k \in \mathbb{R}^n, u_k \in \mathbb{R}^p, y_k \in \mathbb{R}^m$ are the state

vector, the input vector and the output vector at time k respectively; p is the parameter referring to the θ and γ in equations in previous sections.

$\boldsymbol{X}_0$ is the *a priori* known set which contains the unknown initial condition x_0; and $\boldsymbol{P}$ is the set of possible parameters values. The estimation problem under set-membership context consists in finding the unmeasurable state or unknown parameter without losing any possible solution derived from uncertainty. This is also called "guaranteed estimation".

Monte Carlo method Firstly, there exists a method that is based on selecting a number of samples from the value sets. When the number is large, the result is more accurate, but the computing time is also increasing. Though it is not a guaranteed method, it can still be used to demonstrate the trend of the uncertain system. This is a variation of Monte-Carlo method [Kreinovich 2007a].

To estimate the system state using the Monte-Carlo method, one should take samples in the set of parameters $\boldsymbol{P}$ to define the systems with parameter $p \in \boldsymbol{P}$. Consequently the system can be solved by conventional estimation methods. The final solution is the union of all the solutions from systems with parameters $p \in \boldsymbol{P}$.

Take a simple system for example,

$$\begin{cases} x_{k+1} = \begin{bmatrix} 1 & \boldsymbol{h} \\ 0 & 1 \end{bmatrix} x_k + w_k, \\ y_k = \begin{bmatrix} 1 & 0 \end{bmatrix} x_k + v_k, k = 0, 1, 2, ... \end{cases}$$

where $\boldsymbol{h} = h_0 + \triangle h$, with $h_0 = 0.01$ et $\triangle h = [-0.01, 0.01]$, w_k and v_k are known noises in the Equation (3.1).

$$E\{x_0\} = \begin{bmatrix} x_{01} \\ x_{02} \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix},$$
$$Cov\{x_0\} = \begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix} = \begin{bmatrix} 0.5 & 0.0 \\ 0.0 & 0.5 \end{bmatrix},$$
$$Q = \begin{bmatrix} q & 0 \\ 0 & q \end{bmatrix} = \begin{bmatrix} 1 & 0.0 \\ 0.0 & 1 \end{bmatrix}, \; R = r = 10.$$

The result is illustrated in the figure 1.3 with 50 samples distributed averagely within the parameter set. It is easy to see that the number of samples impacts the execution time and the completeness.

The Monte Carlo method is a way to approximate the estimation result under the parameter uncertainties. It can provide us with a tendency of how the set of result tends to be.

1.4.1 Parameter set-membership estimation

The parameter estimation, also known as system identification problem, is an important part of the estimation problem. Although our work concentrates on state estimation,

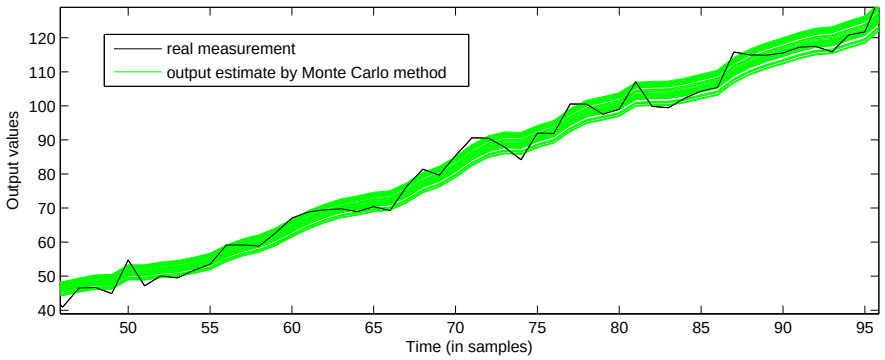

Figure 1.3: Monte Carlo method for estimation problem.

the both estimations share some new advances and techniques in the set-membership context so that it is worth mentioning parameter estimation. A complete introduction of conventional parameter identification and set-membership parameter estimation technique can be found in [Rao 1976, Walter 1997a] and [Walter 1990] respectively.

The guaranteed estimation problem consists in determining the set of the values of the model parameters that are consistent with experimental measurements, where all the errors are assumed to be included in predefined boundaries.

Considering the continuous non-linear parametric state model described by the Equations (1.10), the parameters which need to be estimated belong to $\boldsymbol{P}_0$. The *a priori* known search set contains all the admissible values of the parameter vector p.

To estimate the system parameters under the bounded error context, one should determine the set $\mathbb{P}$ of all the parameters p contained in the initial search set $\boldsymbol{P}_0$. The error between the experimental data and the model output, noted by:

$$\boldsymbol{e}_{p,k} = y_{m,k} - \boldsymbol{y}_{p,k},$$

must belong to the set of admissible error $\mathbb{E}$, where $y_{m,k}$ stands for the output measurement at time k. The $\boldsymbol{e}_{p,k}$ means that this is an interval variable, which is bounded by its upper value $\overline{\boldsymbol{e}_{p,k}}$ and its lower value $\underline{\boldsymbol{e}_{p,k}}$, ; the same definition is applied to $\boldsymbol{y}_{p,k}$. The exact definition can be found in Chapter 2 of this work.

$$\boldsymbol{P} = \{p \in \boldsymbol{P}_0 \mid e(p) \in \boldsymbol{E}\} = \{p \in \boldsymbol{P}_0 \mid \underline{\boldsymbol{e}_{p,k}} \leqslant e_{p,k} \leqslant \overline{\boldsymbol{e}_{p,k}}\}.$$

The characterization of the set $\boldsymbol{P}$ can be defined as a set inversion problem:

$$\boldsymbol{P} = e^{-1}(\boldsymbol{E}) \cap \boldsymbol{P}_0.$$

A guaranteed enclosure of $\boldsymbol{P}$ can be obtained with interval analysis and the Set Inversion Via Interval Analysis (SIVIA) algorithm developed by Luc Jaulin [Jaulin 1994] which are presented in next chapter.

When the system is described by ordinary differential equations, there exists several integration methods for evaluating the model outputs. These methods compute a feasible solution set (FSS) of parameters. This feasible set is consistent with the model structure, obtained measurements and the considered uncertainty. Set-membership approach can be used to identify systems with parametric and nonparametric uncertainties. For example, the interval Taylor model produces the guaranteed numerical integration for system identification in [Ramdani 2005a]; a bounded error identification is done by zonotopes approximation in [Bravo 2004]. Similar works can be found in [Malti 2010, Ramdani 2005c, Chisci 1998].

1.4.2 State set-membership estimation

The conventional state estimation problems are usually solved by probabilistic methods [Walter 1997a] when the noises and perturbations can be reasonably assumed to be random variables. The optimization is used after the choice of an appropriate criterion. These methods rely on the knowledge of a dynamic model of the system under consideration in order to provide an optimal performance. In many cases, however, only an uncertain model of the system is available. In such situations, standard methods fail to provide a guaranteed performance under the presence of parameter uncertainty.

The state estimation based on bounded uncertainty assumption is called set-valued observer [Shamma 1999] or guaranteed state estimator. If the noises are ignored in the state and observation equations while the parameter uncertainty remains, there exists already several works such as [Ploix 1999, Raissi 2004]. [Puig 2003] talks about the solution of worst-case simulation by determining the interval hull enclosing the system states region at every iteration through optimization in a discrete linear time-invariant interval dynamic systems. Linear programming based on state estimation also has its set-membership extension in [Rami 2008] under the same context. Other works which benefit the set-membership uncertainty include [Zhuk 2008, Zhuk 2007]. In some other works the noise is modeled as bounded value without statistical properties; for example. In [Combastel 2003] an observer based on zonotopes is introduced under such context. In [Yang 2009] the noises are assumed to be confined to specified ellipsoidal sets. We can find similar approaches in [Garulli 1999, Hanebeck 2001].

Non-linear system is more difficult. In fact, set integration has been used only recently for nonlinear bounded-error parameter and state estimation with nonlinear continuous time models. There are mainly two kinds of approaches which are used for reconstructing unknown state.

The first approach addresses the case of continuous-time state estimation from discrete-time measurement. It relies on a two-stage methodology: a prediction stage, which consists in reconstructing the state vector by using set integration; a correction stage where are pruned the parts of the predicted state or parameter vector inconsistent with experimental data; for example, in the paper [Durola 2008], authors have proposed a technique for ro-

bust filtering/prediction of nonlinear discrete-time systems with rational dependence on the state and uncertain parameters in a set-membership context, where the prediction is based on a recursive estimation of confidence ellipsoids enclosing the state vector at each sampling time.

The second approach addresses the case of continuous-time state estimation from (almost) continuous-time measurement. The main idea is to build a closed loop interval observer which takes into account measurement errors and model's parameters uncertainties. Two point observers are built which reconstruct the lower and upper bounds of the feasible set for the unknown state or parameter vector when all the bounded uncertainties are considered. This idea has been introduced in [Gouzé 2000, Rapaport 2003, Bernard 2004].

Several approaches have been proposed in the literature to deal with bounded uncertainty ordinary differential equation (ODE) models. In particular, we can distinguish three classes of approaches: those based on interval analysis, those using zonotopes and those using ellipsoids. Among interval based approaches, we can mention three main ones:

Interval Taylor series approximation methods The major problem when dealing with interval methods is the explosion of the size of the enclosures at times $t_0, t_1, ..., t_N$. Consistency techniques allow one to prune the successive enclosures by extending non-linear constraint programming in an interval framework [Deville 2002]. Different filtering operators based on relaxation of the ODE can be defined to prune the search space. They determine whether a box can or cannot contain a solution of the equation. Filters may be based on backward computation, implicit methods, polynomial interpolation [Janssen 1999] or Hermite theorem [Nedialkov 1999a].

Consider the function f which is assumed to be at least k-times continuously differentiable in the real numbers domain $\mathbb{R}^n$. Interval arithmetic is used to compute guaranteed bounds for the solution at the sampling times $(t_1, t_1, ..., t_N)$. The most known methods to solve such a problem are based on Taylor expansions [Rihm 1994, Moore 1966, Nedialkov 1999b]. These methods consist in two parts: they first verify existence and uniqueness of the solution using the fixed point theorem and the Picard-Lindelof operator and compute an a priori enclosure. In the second part, the solution is computed using a Taylor expansion.

In practice, numerical implementation must be done with care. Mean value forms and matrix preconditioning is used in order to control the conservatism, i.e. the spurious uncertainty introduced by interval computation (wrapping and dependence effects).

Zonotopes based estimation methods Zonotopes are a special class of polytopes defined as the linear image of an hypercube. The related linear application is not necessarily invertible. To illustrate this, one can imagine a 3-dimensional cube that would be arbitrarily oriented, then reduced or inflated along some directions and finally projected on a 2-dimensional space (i.e. a plane). Zonotopes are thus convex polytopes with cen-

tral symmetry that can be implicitly represented by a simple matrix describing the linear application transforming a unit hypercube (i.e. an aligned box bounded by the interval $[-1,\ +1]$ along each elementary direction) into the related zonotope. One of the interest of zonotopes consists in using an implicit and compact matrix representation to represent some polytopes with a possibly very large number of vertices and facets. As many zonotope operators can be reduced to matrix operations, no costly enumeration of vertices and facets is required in most cases. For instance, the Mikowski sum of two zonotopes can be computed by a matrix concatenation; the linear image of a zonotope can be computed by a matrix product, etc. Moreover, zonotopes constitute a rather natural way to represent the reachable set of linear time varying discrete-time dynamical systems with a good control of the so-called wrapping effect [Kühn 1998]. The trade-off between precision and computational load can be tuned through the parameterization of an operator allowing to reduce the complexity of a given zonotope at the price of some outer approximation of the initial domain. Even if the intersection between two zonotopes is not, in general, a zonotope, it is however possible to design some algorithms based on standard linear algebra to compute an outer approximation of the intersection between two zonotopes. Following the principle of bounded error state observation, this can be used to design set-membership observers based on a prediction/correction strategy [Combastel 2003], like Kalman filters in a stochastic context. The case of non-linear systems is treated by a linearizion of the system around the center of the currently computed domain and an inclusion of the linearization error. Parametric uncertainties are taken into account either using interval matrices and so-called notion of "zonotope family" introduced in [Alamo 2003] or by a state extension like in [Combastel 2004] and [Combastel 2005]. The former is interesting from the complexity point of view (the state dimension remains unchanged) but suffers from some kinds of conservativeness as the dependency between uncertain physical parameters is not well captured. The latter (state extension) allows to take such dependencies into account but may induce many computations if the number of uncertain parameters is not small enough. However, it can be noticed that zonotopes are well-adapted not only to deal with small state dimensions which is also one of their interest compared to other set-membership algorithms, especially when bisections are involved.

Clearly, there has been recently a significant progress in bounded error estimation methods with continuous time models which makes plausible their future use for fault detection and isolation.

Robust filter We should also mention that there exists a large research field on "robust filter", where the system parameters are assumed to be time-varying and "norm-bounded". The major trade-off such as Kalman filter design is between the performance and robustness when uncertainty is present [Xu 2009]. The basis is to design a stochastic stable quadratic state estimator such that the estimation error covariance has a guaranteed bound for all admissible uncertainties but not the state itself. Related work can be found in [Shi 1999, Petersen 1999, Xie 1994, Zhu 2000, Souto 2008, Ra 2008].

1.5 Application to fault detection

Modern systems are designed to accomplish certain tasks automatically, in order to reduce the human interventions. But the reliability of automatic components, sensors, actuators are so crucial that it needs a surveillance system able to detect the faults when they occur, locate the faulty component and if it is possible, restore the system to normal operating conditions.

1.5.1 Fault modeling

We consider a system as a mechanism which can accomplish certain tasks with predefined inputs and desirable outputs. Modern technology considers operational safety by evaluating the behaviour of the system. A deviation from normal behavior is possible in any process. If the deviation is foreseeable and acceptable, it is called uncertainty; but if the deviation is unexpected or inordinate, it may have a serious impact on equipment, production quality, security, economy, levels of contamination; in the worst case a fault may cause fatal accidents. The development of fault detection and diagnosis tools to help isolating the abnormal behavior during operating processes or off-line is a very active area in automation.

Definition 1.1 *A **fault** [Isermann 1997a] is an unpermitted deviation of at least one characteristic property or variable of the system.*

The faults in the system are generally classified into three types: actuator faults, sensor faults, and component faults. For example on linear system, the faults can be formally modeled as :

$$\begin{cases} \dot{x}(t) = (A + \triangle A)x(t) + (B + \triangle B)u(t) + E_1 n_1(t), \\ y(t) = (C + \triangle C)x(t) + (D + \triangle D)u(t) + E_2 n_2(t), \end{cases} \tag{1.11}$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $u(t) \in \mathbb{R}^p$ is the input vector, $y(t) \in \mathbb{R}^m$ is the output vector measured by the sensors. A, B, C, D represent respectively state matrix, input matrix, output matrix and feed-forward matrix. We assume that the input vector $u(t)$ is known, but the system model may contain model uncertainties $\triangle A$, $\triangle B$, $\triangle C$, $\triangle D$, noise or unknown disturbance vectors n_1, n_2 and their matrices correspond to E_1, E_2.

A component fault may happen when there exists an extreme change in the circumstance of the process feed stream or ambient conditions. For example, when there is ice accumulated on the wings of a plane, the parameters of the dynamic system are different than designed and the plane may probably respond to the control in an unpredicted way.

An actuator problem occurs when it can not provide desired command signals to conduct the system behaviour properly.

A sensor fault can be very tricky as the measurements are produced with unpredicted bias. It is important to distinct whether it is the measurement noise or a sensor error, or even the presence of both.

To ensure the operations, the system must satisfy the performance specifications, the faults in the process should be detected, diagnosed and eliminated.

1.5.2 Fault detection problem

To design a surveillance system, different approaches are possible. The available information for the surveillance system and the foreseeable type of faults play an important role in the choice of the approach.

Two main approaches can be mentioned:

- **Material redundancy:** the technique of material redundancy is quite intuitive. The decision is made by voting system composed of redundant sensors. The voting result from multiple sensors can take control of the functions, or identify the faulty component. However, it adds complexity to the hardware design and general cost.
- **Analytical redundancy:** it is based on on the availability of a mathematical model. This approach consists in comparing the measurements from the real system and the values "predicted" by the system model. The difference between them is called *residual*.

The fault indicators can be built from the different approaches. By monitoring the set of fault indicators, fault alarms can be trigged. The collection of fault indicators can be used to proceed fault localization and identification. Once the fault is isolated, actions can be deployed to restore the system in a normal state. A classical system supervision procedure can be presented as in Figure 1.4.

Analytical redundancy, also known as model-based fault diagnosis, is the approach chosen in our work. We consider set-membership state estimation methods for setting the fault alarm thresholds. The scenario of analytical diagnostic is shown in Figure 1.5.

The estimation is an important step of the entire procedure as it participates to the residual generation. Our goal is to provide a reliable estimate under the presence of multi-type uncertainties in order to detect faults correctly, effectively and rapidly.

In the literature, Willsky introduced key points of analytical redundancy in model-based fault detection in a survey paper [Willsky 1976] early in 1976. In 1981 Desai presented a general methodology of fault detection [Desai 1981]. In [Isermann 2005, Isermann 2006, Muenchhof 2009, Isermann 1997b, Isermann 1997a] the authors present applications using fault detection techniques. [Staroswiecki 2001] extends the analytical redundancy techniques to FDI in non-linear dynamic systems. Three major methods can be outlined from these works: the *parameter estimation approach*, the *parity relation approach* and the *observer based method*.

Based on abundant work in the field, it is interesting to overview the development and progress of fault detection techniques by using set-membership estimation. Our work concentrates on the system state reconstruction in the set-membership context; nevertheless all the methods are important to explore as a reference for comparison.

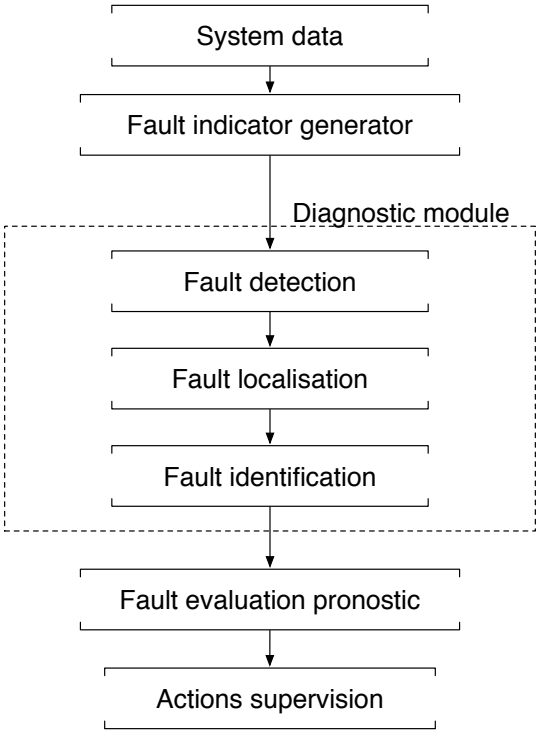

Figure 1.4: System supervision and conduit procedure

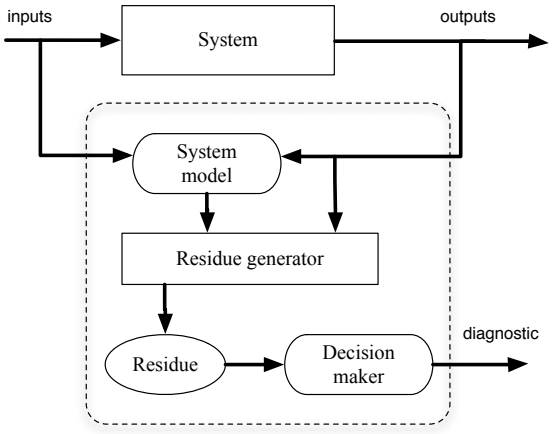

Figure 1.5: Diagnostic by analytical methods

1.5.2.1 Model-based fault diagnosis

The basic concept of analytical redundancy-based fault diagnosis is about residual generation as it is shown in Figure 1.6.

A residual $r(t)$ can be obtained by a generation function g which depends on the input vector $u(t)$ and the output vector $y(t)$:

$$r(t) = g(u(t), y(t)), \tag{1.12}$$

This implements the difference between the measured output $y_m(t)$ and the estimated output $\hat{y}(t)$ estimated from a mathematical model:

$$r(t) = y_m(t) - \hat{y}(t). \tag{1.13}$$

In the set-membership context, the residual generation should be modified to suit the presence of uncertainty. Set estimation algorithms have drawn lots of attention in the fault detection domain since the 90's. Abundant methods are based on the assumptions from the functional form of the regression function describing the relations between measured input and output, where the noise is assumed to be bounded. According to this assumption, the set-membership approach seems to be quite promising and a good alternative to either linear or non-linear, statical or dynamical, deterministic or uncertain systems [Raissi 2004]. It has become a powerful tool in many fields. Fault diagnosis algorithms can benefit from the ability of set-membership algorithms to describe sets of trajectories. It is indeed important not only to compute a single (nominal) trajectory but also to quantify the uncertainty around it in order to design efficient detection and diagnosis tests. The last decade has seen several publications in that field where zonotopes are used to extend model-based residual generation methods like parity space [Ploix 2001, Lalami 2006], observers [Puig 2003, Stancu 2005] and parameter estimation.

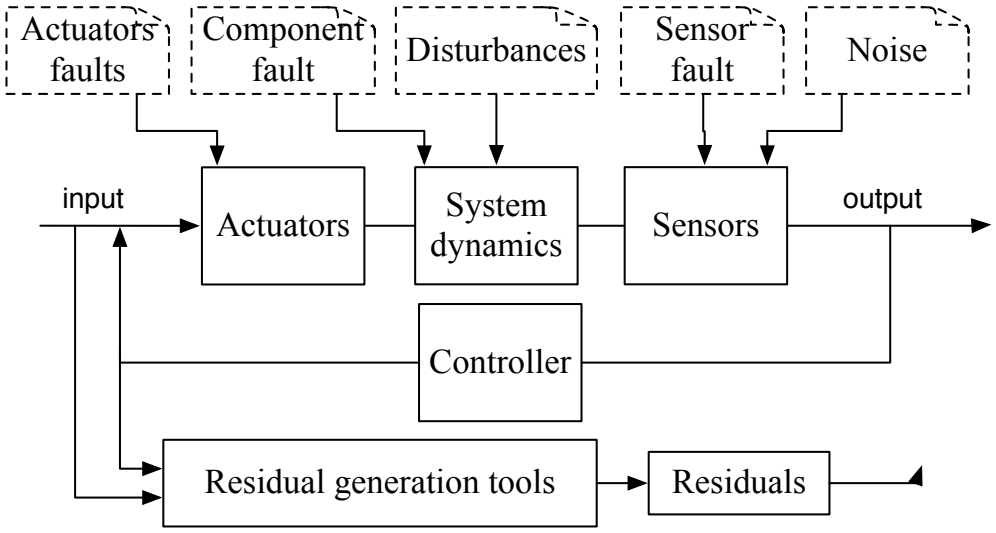

Figure 1.6: System and residual generation.

In the next subsections, we present the three major approaches for model-based fault diagnosis and their advances to date in the set-membership context.

1.5.2.2 Parameter Estimation Approach

Parameter estimation used in the fault detection problem is based on the assumption that system faults are reflected in a variation of the parameters of the system model re-

gardless of their multiplicative or additive nature. The fault detection problem is then formulated as detecting any change in the system parameters.

The general procedure to detect faults with the parameter estimation approach has been summarized in [Escobet 2001]. A normal behaviour system is modeled by an input/output mathematical form as follow:

$$y(t) = f(x(t); u(1), ..., u(t); n_2(t), \gamma), \tag{1.14}$$

with the same notations as Eq.(1.11), while γ represents the non measurable parameters related to faults.

By using a suitable estimation function f_γ, the residual $r(t)$ can be obtained from comparing the estimated parameters $\hat{\gamma}$ and model parameters:

$$\hat{\gamma}(t) = f_\gamma(y(1), \cdots, y(t), u(1), \cdots, u(t)), \tag{1.15}$$

$$r(t) = \mid \gamma(t) - \hat{\gamma}(t) \mid . \tag{1.16}$$

Based on the residual and tolerance limits, a binary decision can be made on the system status anticipating predetermined threshold level or using statistical decision theory [Isermann 1984].

Parameter estimation has been well developed since years [Bellman 1971, Ljung 1999]. It is a very mature field in control theory [Gevers 1996], naturally related to probability theory. A complete presentation can be found in various books especially [Walter 1997b]. It covers uncertainty management with Monte-Carlo methods and bounded-error set estimation. [Walter 1997b] also briefly discussed how to test normality by using parameter estimation. In an early survey paper [Isermann 1984], Isermann illustrated how process fault diagnosis can be achieved via estimation of unmeasurable process parameters and/or state variables. Simani and al. wrote a book [Simani 2003] on fault detection methods focusing on system identification techniques.

Parameter estimation Approach in the Set-membership Context When the model output is linear w.r.t the parameters, it is possible to use different types of algorithms. There exists the ellipsoid-based approach which produces ellipsoidal outer bounds for the feasible parameter set. The involved computations are recursive, very limited and similar to those required by recursive least squares. However, the set estimates of the uncertain parameters are often very pessimistic. As complementary, the orthotopic approximations algorithm can be used [Walter 1990]. The computations involved are not recursive but more complex, however the parameter uncertainty set can be obtained with more accuracy and used for further purpose.

[Ploix 1999] discussed static linear models characterized by bounded uncertainties in both the equation error and the parameters. This work proposed an algorithm for characterizing uncertainties. The author claims that in certain cases the coexistence of parameter uncertainty and measurements uncertainty is difficult to take into account at the same time due to the potential interaction.

A recent work presented by Ari Ingimundarson and al. in [Ingimundarson 2009] shows that consistency checks indicating faults of linear systems can be performed in a natural manner with a zonotope description of the feasible parameter set. According to this work, the zonotopes provide better estimates of the parameter set, leading directly to a better trade-off between false alarm and missed alarm. The main contribution is to present a robust fault detection algorithm for a regression equation that uses zonotope estimates of the parameter uncertainty set and is based on an efficient algorithm presented in [Bravo 2004].

The parameter estimation problem of fractional models based on frequency domain uncertain but bounded data is discussed in [Malti 2010]. The set inversion technique is used to compute the set of all the parameter vectors in order to verify if the model output is consistent with the measured data and the error bounds. Two methods based on ICSP (interval Constraint Satisfaction Problem) are used and compared. The first one formulates a real-CSP (Constraint Satisfaction Problem) based on an explicit decomposition of the frequency response in the real and the imaginary parts; the second one formulates a complex-CSP using the complex frequency response without any decomposition. According to this work, to reduce pessimism and computation time.

An interval based fault detection application of parameter estimation approach on aerospace field can be found in [Bouron 2002, Pons 2008].

1.5.2.3 Parity Relations Method

The Parity relations approach is a direct implementation of the concept of "redundancy". Primary residuals are formed as the difference between the actual outputs and those predicted by the model.

The model is based on the state-space framework as Equation(1.11). $f_a(t)$ is the actuator fault vector and $f_s(t)$ is the sensor fault vector.

$$\begin{cases} \dot{x}(t) = Ax(t) + Bf_a(t), \\ y(t) = Cx(t) + f_s(t). \end{cases} \tag{1.17}$$

$p(t)$ is defined as a parity vector to fulfill:

$$p(t) = Vy(t) = VCx(t) + Vf_s(t). \tag{1.18}$$

By choosing the matrix V such that $VC = 0$, we have:

$$p(t) = Vf_s(t). \tag{1.19}$$

The ith column of V determines the direction of the parity vector $p(t)$ due to the ith sensor fault. This method is known as the Chow Willsky scheme [Chow 1984], or the parity space approach.

Contrary to the principle of the observer-based approach which is to estimate the unknown variables and the state variable, the principle of the parity relations approach is

to eliminate the unknowns to derive relations that only involve known variables (parity relations). There are various schemes to formulate residual generation by using parity relations. In general, the residual generation filters should be designed to enhance fault isolation so that they each exhibit directional or structural properties in response to a particular fault and they also need to maintain robustness to noise, disturbances, or model errors. In [Gertler 1997], the basic concepts of residual generation for both additive and multiplicative faults are developed. J. Gertler also reviewed the link between the parity relations method and the other two major approaches, observer-based diagnosis and parameter estimation.

Parity Relations Approach in the Set-membership Context Since parity equations are an obvious method to implement residual generators, there are many parity relation approaches in which uncertainties are handled in a set-membership context.

In [Adrot 2008], the parity space approach is used for generating testable redundancy relations in which each uncertain parameter is defined by an interval containing all its feasible values, and the solution domain is approximated by a convex zonotope.

Since the parity space approach leads to mathematical relations with a finite time horizon and time-invariant uncertain parameters over this time horizon, a proper determination of horizon size is a key issue in parity relations approach in set-membership context. In [Adrot 2000], residuals are linearized in the bounded variables and a reduction procedure is applied to decrease their complexity. Parity space approach is chosen to avoid wrapping effect from interval analysis in case of set-membership recursive models. It reformulates the dynamic model equations in the form of algebraic relations on a chosen time horizon. By stacking sensor observations on the time window according to initial state vector, a static representation can be obtained. The importance of the horizon determination is supported in that work.

Calderón-Espinoza and al. tested the consistency between the interval model and the real process using interval measurements in [Calderón-Espinoza 2007b], by including bounded uncertainties of the sensors. To compute the envelope limits, an iterative computation of the external and the internal estimations of the exact envelope is made by a branch-and-bound algorithm.

In [Calderón-Espinoza 2007a] the diagnosis with set-membership parity relation approach is discussed. By the side of real application, in [Ploix 2006], a new approach for the design of parity relations for linear dynamic systems with additive and multiplicative uncertainties is presented and applied on a two water tank application. Deterministic parity relations are replaced by uncertain parity relations to compute detection decisions in the form of a set of inequalities to be satisfied. At each sampling time, the final numerical expression is recomputed by using new income measurement data. The consistency is checked by computing and testing the constraints.

1.5.2.4 Observer Based Methods

The observer based approach generates the residual from the state estimate and output based on Eq (1.13).

For example, Let $\hat{x}(t)$ represent estimated state and $\hat{y}$ represent the estimated output, a full state observer is then written as:

$$\begin{cases} \hat{x}(t+1) = A\hat{x}(t) + Bu(t) + L(y_m(t) - \hat{y}(t)), \\ \hat{y}(t) = C\hat{x}(t), \end{cases} \tag{1.20}$$

where L is the gain matrix, $y_m(t)$ is the measurement at instant t. In this case, the component faults are assumed to be zero, the residual can be given as:

$$r(t) = W(y_m(t) - \hat{y}(t)) = WC\xi(t), \tag{1.21}$$

where W is a weighting matrix and ξ is the state estimate error. The purpose here is to choose L and W such that the residual $r(t)$ has some desired sensitivity properties concerning certain faults.

As we know, the observer based fault detection method is based on a system model, obtained by commonly applying some approximation and linearizion during the system modeling. In addition to disturbances and noises in the practical engineering, uncertainties can drive the system insensitive to the faults, or activate false alarm.

Therefore, in the literature, lots of work have been done to improve robustness of the observer-based residual generators. D. Luenberger had a very complete coverage of state observing techniques over linear systems in early days [Luenberger 1964]. From the 90s, an adaptive observer/filter has been introduced [Frank 1994], [Wang 1997], which can be seen as a combination of observer and parameter based method. This method can handle some uncertainties, and non-linearities as well so that the adaptive observer, its robust analysis and optimization provide a very important track of the literature.

In the survey [Garcia 1997] some schemes are discussed to extend the well-known diagnosis methods for linear systems to the non-linear case which is also a very important direction in this domain.

Observation Based Methods in the Set-membership Context State observer based fault diagnosis is as popular as parameter estimation approach.

A set-valued observer (also called guaranteed state estimator) produces a set of possible states based on output measurements and models; this is a direct interpretation of observation based methods in the context of set-membership. This technique consists in checking whether the measurements of the output belong to the set of all possible estimated outputs obtained considering uncertainty on model parameters [Stancu 2005].

As early as in the 90s, in the survey [Frank 1997], the membership value based estimation plus the fuzzy threshold logic are used to deal with the noises and disturbances in an observer based residual generator and fault detection. Thresholds are tweaked as small

as possible because any increase of the threshold is associated with a loss of sensitivity to faults. The interval with a properly chosen upper and lower bounds and membership functions defining faulty(non-zero) and non faulty(zero) is considered in a fuzzy sense.

The interval observer discussed in [Rami 2008] is a Linear Programming approach to the problem of designing observers that ensure guaranteed bounds on the estimated states when uncertainties are present. A new type of robustness observers involving dilatation functions is introduced to ensure robustness subjected to uncertainties and non-linearities. The state estimator for non-linear dynamical systems is presented in [Marx 2010, Raissi 2004] where the noise and perturbation are assumed bounded. Some applications in the field of aeronautic test bench and data validation are presented in [Alhaj-Dibo 2004, Karim 2008, Jauberthie 2009].

1.6 Conclusion

Until now, we have reviewed basic notions for dealing with uncertainty, the estimation problem and faults. Different formalizations of the uncertainties are used for different cases. Several fault detection techniques are presented, along with their recent advances in the set-membership context.

As it is shown in the overview, works generally make the hypothesis that the statistical uncertainties (errors and noises) and the bounded uncertainties (parameter variation, measurement tolerances) are considered separately. In real applications, these uncertainties usually coexist. Our work consists in providing a complete structure for fault detection in the set-membership context, in which some uncertainties are modeled as bounded sets, and noises are model led statistically at the same time. This work concentrates on a state estimation method and the related fault detection problem. Conventional approaches are foundations of their set value extension but in the set-membership context the estimation meets particular challenges.

Chapter II
Interval analysis

CHAPTER 2

Interval analysis

Contents

2.1 Introduction

System modelling can be difficult due to the presence of noise and perturbations. When the model structure and uncertainty characteristics are known, the process state can be reconstructed. Under the stochastic approach, the estimation problem depends on the statistical distribution of the noises and perturbations. In practice, there are many cases in which the statistical distribution is not known.

An alternative approach consists in considering uncertainty differently. The only available information on uncertain variables is that their value belongs to a known bounded set. This is the principle of the set-membership approach. The sets are characterized by their boundaries. The uncertain magnitudes are assumed to belong to known acceptable bounds. The results obtained by set-membership estimations provide all the solutions consistent with the predefined uncertainties.

As shown previously, set-membership estimation can characterize a set of state or parameter vector values which are admissible, namely those resulting in trajectories which are within a priori given error bounds. Different geometrical forms of sets are possible, from ellipsoids to zonotopes, but also boxes as used in this thesis.

Interval analysis was originally introduced by R. E. Moore[Moore 1966]. It is particularly used for system robust control [Kreinovich 2011, Rauh 2009, Dao 2006] and for estimation tasks on which we are working [Poignet 2003a, Ramdani 2005b]. It also has numerous applications in pure mathematical problems [Janssen 2003, Kamali 2011] or computer science [Kreinovich 2007b, Melquiond 2008]. Recently, some attempts have been made to improve the interval arithmetic, such as the generalized interval arithmetic, which is developed to solve the dependency problem existing with the traditional interval arithmetic [Wang 2008b], but its implementation in real applications has not yet been proven to be more accurate or efficient than the traditional arithmetic.

In this chapter, the basic concepts and definitions of interval analysis are presented. Most of the notions of interval analysis can be found in [Jaulin 2001d]. The new advances for improving the pessimism of interval solutions are introduced in Section 2.3, especially the usage of constraint propagation techniques; in Section 2.4 we emphasize the problem of interval matrices inversion. A new approach to obtain the matrix inversion by using a set inverse algorithm is proposed, followed by a brief introduction of the toolbox that we uses for simulation and validation purposes.

2.2 Basic definitions

Definition 2.1 *(Interval) An interval $\boldsymbol{x}$ in $\mathbb{R}$ is a closed set of connected real values noted by*

$$\boldsymbol{x} = [\underline{\boldsymbol{x}}, \overline{\boldsymbol{x}}] = \{x \in \mathbb{R} | \underline{\boldsymbol{x}} \leqslant x \leqslant \overline{\boldsymbol{x}}\}.$$

where $\underline{\boldsymbol{x}} \in \mathbb{R}$ is the inferior bound of interval $\boldsymbol{x}$ and $\overline{\boldsymbol{x}}$ is the superior bound.

The set of all intervals in $\mathbb{R}$ is noted as $\mathbb{IR}$.

In our work, the interval variable is represented in **bold type** as in [Kreinovich 2007b] for example. There exist also other ways to represent an interval. Some use superscript I as x^I in [Chen 1997] to represent an interval, others use square brackets as $[x]$ as in [Jaulin 2001a].

Different expressions for bounds exist as $[x_{inf}, x_{sup}]$, $[x^-, x^+]$([Jaulin 2006]). The superior bound and inferior bound can also be obtained by operators $sup(\bullet)$ and $inf(\bullet)$ respectively.

Definition 2.2 *Given $\boldsymbol{x} \in \mathbb{IR}$, we define:*

$$\begin{aligned} &\textit{the interval width}: && w(\boldsymbol{x}) = \overline{\boldsymbol{x}} - \underline{\boldsymbol{x}}, \\ &\textit{the interval center}: && mid(\boldsymbol{x}) = (\overline{\boldsymbol{x}} + \underline{\boldsymbol{x}})/2, \\ &\textit{the interval radius}: && rad(\boldsymbol{x}) = (\overline{\boldsymbol{x}} - \underline{\boldsymbol{x}})/2. \end{aligned}$$

The interval width defines the grade of uncertainty of variable. There exists several different forms of representations, such as the equivalence of $mid(\boldsymbol{x})$ to x_c and $rad(\boldsymbol{x})$ to Δx.

The difference between $rad(\boldsymbol{x})$ and the Δx is that the latter is actually a random value from $[-rad(\boldsymbol{x}), rad(\boldsymbol{x})]$, while the former is a determined value. It is possible to encounter a different form of interval definition:

$$\boldsymbol{x} = x_c + \Delta x, \tag{2.1}$$

which is also widely used in applications. In this formulation, Δx can not be replaced by $rad(\boldsymbol{x})$ and mixed with x_c as $rad(\boldsymbol{x})$ is a punctual value while Δx is actually a set of values.

Real values, that we also qualify as "punctual", can be easily represented in an interval framework:

Proposition 2.1 *A punctual value x in $\mathbb{R}$ can be seen as an interval value $[x]$ as follows:*

$$x = \boldsymbol{x} = [\underline{\boldsymbol{x}}, \overline{\boldsymbol{x}}] = \{x \in \mathbb{R} | \underline{\boldsymbol{x}} \leqslant x \leqslant \overline{\boldsymbol{x}}, \underline{\boldsymbol{x}} = \overline{\boldsymbol{x}}\}.$$

Then we have $w(\boldsymbol{x}) = 0$, $rad(\boldsymbol{x}) = 0$, $mid(\boldsymbol{x}) = \overline{\boldsymbol{x}} = \underline{\boldsymbol{x}}$. Punctual variables are sometimes called *degenerated interval* ([Chabert 2009, Markov 1996, Wang 2007, Shevtshenko 2009, Goldsztejn 2005, Goldsztejn 2007]).

Definition 2.3 *(Interval vector) An interval vector noted by:*

$$\boldsymbol{x} = (\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n)^T,$$

is a vector in which the elements are intervals. $\boldsymbol{x} \in \mathbb{IR}^n$ which is the set of interval variables in $\mathbb{IR}$. For all $i \in 1, ..., n$, the interval $\boldsymbol{x}_i$ corresponds to the i_{th} component of $\boldsymbol{x}$.

$\boldsymbol{x}_i$ can also be noted as $\boldsymbol{x}^i$ in case the subscript place is used for other index such as discrete time k, under the condition of proper advanced statement in place.

Definition 2.4 *(Interval matrix) An interval matrix noted by*

$$\boldsymbol{X} = \begin{pmatrix} \boldsymbol{x}_{1,1} & \cdots & \boldsymbol{x}_{1,m} \\ \vdots & \ddots & \vdots \\ \boldsymbol{x}_{n,1} & \cdots & \boldsymbol{x}_{n,m} \end{pmatrix},$$

is a matrix in which the elements are intervals; $\boldsymbol{X} \in \mathbb{IR}^{n\times m}$ which is the set of interval vectors of $\mathbb{IR}^n$. For all $i \in 1,...,n$ and $j \in 1,...,m$, the interval $\boldsymbol{x}_{i,j}$ corresponds to the i_{th} row j_{th} column component of $\boldsymbol{X}$.

$\boldsymbol{x}_{i,j}$ can also be noted as $\boldsymbol{x}^{(i,j)}$ in case the subscript place is used for other index such as discrete time k, under the condition of proper advanced statement in place.

The basic concepts defined for intervals are also defined for interval vectors and matrices.

Definition 2.5 *If $\boldsymbol{x} \in \mathbb{IR}^n$, we define:*

$$\begin{aligned}
&\textit{the interval vector inferior bound}: \quad \underline{\boldsymbol{x}} = inf(\boldsymbol{x}) = (\underline{\boldsymbol{x}}_1, \underline{\boldsymbol{x}}_2, ..., \underline{\boldsymbol{x}}_n)^T, \\
&\textit{the interval vector superior bound}: \quad \overline{\boldsymbol{x}} = sup(\boldsymbol{x}) = (\overline{\boldsymbol{x}}_1, \overline{\boldsymbol{x}}_2, ..., \overline{\boldsymbol{x}}_n)^T, \\
&\textit{the interval vector center}: \quad mid(\boldsymbol{x}) = (\overline{\boldsymbol{x}} + \underline{\boldsymbol{x}})/2, \\
&\textit{the interval vector radius}: \quad rad(\boldsymbol{x}) = (\overline{\boldsymbol{x}} - \underline{\boldsymbol{x}})/2, \\
&\textit{the interval vector width}: \quad w(\boldsymbol{x}) = \max_{i=1}^{n} \left(\overline{\boldsymbol{x}}_i - \underline{\boldsymbol{x}}_i\right), \\
&\textit{the interval vector volume}: \quad Vol(\boldsymbol{x}) = \prod_{i=1}^{n} w(\boldsymbol{x}_i) = (\overline{\boldsymbol{x}}_1 - \underline{\boldsymbol{x}}_1) \cdot ... \cdot (\overline{\boldsymbol{x}}_n - \underline{\boldsymbol{x}}_n).
\end{aligned}$$

Notice that interval width and interval volume characterize different aspects of a given interval vector.

Definition 2.6 *If $\boldsymbol{X} \in \mathbb{IR}^{n\times m}$, we define:*

$$\begin{aligned}
&\textit{the interval matrix inferior bound}: \quad \underline{\boldsymbol{X}} = inf(\boldsymbol{X}) = \begin{pmatrix} \underline{\boldsymbol{x}}_{1,1} & \cdots & \underline{\boldsymbol{x}}_{1,m} \\ \vdots & \ddots & \vdots \\ \underline{\boldsymbol{x}}_{n,1} & \cdots & \underline{\boldsymbol{x}}_{n,m} \end{pmatrix}, \\
&\textit{the interval matrix superior bound}: \quad \overline{\boldsymbol{X}} = sup(\boldsymbol{X}) = \begin{pmatrix} \overline{\boldsymbol{x}}_{1,1} & \cdots & \overline{\boldsymbol{x}}_{1,m} \\ \vdots & \ddots & \vdots \\ \overline{\boldsymbol{x}}_{n,1} & \cdots & \overline{\boldsymbol{x}}_{n,m} \end{pmatrix}, \\
&\textit{the interval matrix center}: \quad mid(\boldsymbol{X}) = (\overline{\boldsymbol{X}} + \underline{\boldsymbol{X}})/2, \\
&\textit{the interval matrix radius}: \quad rad(\boldsymbol{X}) = (\overline{\boldsymbol{X}} - \underline{\boldsymbol{X}})/2, \\
&\textit{the interval matrix width}: \quad w(\boldsymbol{X}) = \max_{i=1,j=i}^{n,m} \left(\overline{\boldsymbol{x}}_{i,j} - \underline{\boldsymbol{x}}_{i,j}\right).
\end{aligned}$$

The interval vector $\boldsymbol{x}$ can also be characterized as a Cartesian product of intervals; for differentiating their usages, it is suggested to define explicitly.

Definition 2.7 *(Interval box) An interval box $\boldsymbol{x}$ in $\mathbb{IR}^n$ is the Cartesian product of n intervals:*

$$\boldsymbol{x} = [\underline{x}_1, \overline{x}_1] \times ... \times [\underline{x}_n, \overline{x}_n] = \boldsymbol{x}_1 \times ... \times \boldsymbol{x}_n. \tag{2.2}$$

The interval box has the identical characteristics as an interval vector. Moreover, an operation called *"the bisection"* can divide a box into two boxes.

Definition 2.8 *(Sub-paving) A sub-paving is defined as a union of interval boxes. Particularly, a regular sub-paving is constituted by a set of disconnected boxes who do not share their boundaries.*

Definition 2.9 *(Bisection) The bisection is an operation that partitions an interval box $\boldsymbol{x}$ into two other interval boxes $L(\boldsymbol{x})$ and $R(\boldsymbol{x})$ which are:*

$$L(\boldsymbol{x}) \triangleq [\underline{x}_1, \overline{x}_1] \times ... \times \left[\underline{x}_j, \frac{\underline{x}_j + \overline{x}_j}{2}\right] \times ... \times [\underline{x}_n, \overline{x}_n], \tag{2.3}$$

$$R(\boldsymbol{x}) \triangleq [\underline{x}_1, \overline{x}_1] \times ... \times \left[\frac{\underline{x}_j + \overline{x}_j}{2}, \overline{x}_j\right] \times ... \times [\underline{x}_n, \overline{x}_n], \tag{2.4}$$

where the jth component of $\boldsymbol{x}$ is bisected.

It is clear that we have $\boldsymbol{x} = L(\boldsymbol{x}) \cup R(\boldsymbol{x})$. The bisection is the foundation of most interval arithmetic based algorithms which are destined to solve the set inverse problem.

Generally the to-be-bisected j_{th} component is chosen so that $w(\boldsymbol{x}_j) = w(\boldsymbol{x})$. But other strategies can be adopted for set inversion or global optimization.

2.2.1 Interval arithmetic

The mathematical operations allowing one to manipulate interval variables are presented in this section. See [Moore 1966, Kearfott 1996, Jaulin 2001d] for detailed reviews.

2.2.1.1 Natural arithmetic

The operations on real numbers can be extended to intervals by the following formula:

$$\boldsymbol{x} \diamond \boldsymbol{y} = [\{x \diamond y \mid x \in \boldsymbol{x}, y \in \boldsymbol{y}\}], \tag{2.5}$$

with $\diamond \in \{+, -, \times, /\}$.

Given $\boldsymbol{x}$ and $\boldsymbol{y}$, two intervals in $\mathbb{IR}$, the characterization of $\boldsymbol{x} \diamond \boldsymbol{y}$ can be obtained by:

$$\begin{aligned}
\boldsymbol{x} + \boldsymbol{y} &= [\underline{x} + \underline{y}, \overline{x} + \overline{y}], \\
\boldsymbol{x} - \boldsymbol{y} &= [\underline{x} - \overline{y}, \overline{x} - \underline{y}], \\
\boldsymbol{x} \times \boldsymbol{y} &= [min(\underline{x}\underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{x}\overline{y}), max(\underline{x}\underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{x}\overline{y})], \\
\boldsymbol{x}/\boldsymbol{y} &= \boldsymbol{x} \times 1/\boldsymbol{y},
\end{aligned} \tag{2.6}$$

with:

$$1/\boldsymbol{y} = \begin{cases} [-\infty, \infty] & \textit{if} \quad \boldsymbol{y} = [0,0] \\ [1/\overline{\boldsymbol{y}}, 1/\underline{\boldsymbol{y}}] & \text{if} \quad 0 \notin \boldsymbol{y} \\ [1/\overline{\boldsymbol{y}}, \infty] & \textit{if} \quad \underline{\boldsymbol{y}} = 0, \overline{\boldsymbol{y}} > 0, \\ [-\infty, 1/\underline{\boldsymbol{y}}] & \textit{if} \quad \underline{\boldsymbol{y}} > 0, \overline{\boldsymbol{y}} = 0, \\ [-\infty, \infty] & \textit{if} \quad \underline{\boldsymbol{y}} < 0, \overline{\boldsymbol{y}} > 0. \end{cases} \tag{2.7}$$

Unfortunately, the result of a series of operations between two or more intervals is not necessarily "minimal" in the sense that the interval result may include spurious values. The interval result is therefore pessimistic. The problem has two major sources, multi occurrence or dependence effect and wrapping effect.

2.2.1.2 Dependency effect

Given an interval, $\boldsymbol{x} \in \mathbb{IR}, w(\boldsymbol{x}) \neq 0$, the definition 2.5 gives:

$$\boldsymbol{x} \diamond \boldsymbol{x} = [\{x_1 \diamond x_2 | x_1 \in \boldsymbol{x}, x_2 \in \boldsymbol{x}\}]. \tag{2.8}$$

with $\diamond \in \{+, -, \times, /\}$. According to this equation, the same interval appears more than once in the expression: the natural interval arithmetic considers two different real values belonging to this interval, there is no constraint specifying that $x_1 = x_2$. In comparison, in point wise arithmetic, each occurrence of the same variable is equal to others. This is called the *dependency effect*.

This effect provides several special characteristics for the natural interval arithmetic.

Non existence of zero in subtraction In point-wise arithmetic, zero can be obtained by the subtraction of two identical numbers, while in natural interval arithmetic, the point zero after the subtraction no longer exists,except for degenerated intervals.

Example 2.1 *Given* $\boldsymbol{x} = [2,4]$, *then* $\boldsymbol{x} - \boldsymbol{x} = [-2,2] \neq 0$, *nevertheless* $0 \in (\boldsymbol{x} - \boldsymbol{x})$.

Non existence of unity in division In point-wise arithmetic, unity can be obtained by the division of two identical numbers, while in natural interval arithmetic, the point unity after the division no longer exists,except for degenerated intervals.

Example 2.2 *Given* $\boldsymbol{x} = [2,4]$, *then* $\boldsymbol{x}/\boldsymbol{x} = [0.5,2] \neq 1$, *nevertheless* $1 \in (\boldsymbol{x} - \boldsymbol{x})$.

Constraint loss When identical variable is constructed distinctly in different operations and equations, it is non doubt that the constraint between identical variables can no longer hold.

Example 2.3 *Given* $\boldsymbol{x} = [-1,2], \boldsymbol{y} = [2,3]$, *then* $\boldsymbol{x} \times \boldsymbol{y} \times \boldsymbol{x} = [-1,2] \times [2,3] \times [-1,2] = [-6,12]$.

There exists a constraint. $\mathcal{C}_x = \{\boldsymbol{x}_1 \times \boldsymbol{y} \times \boldsymbol{x}_2 \mid x_1 = x_2, x_1 \in \boldsymbol{x}, x_2 \in \boldsymbol{x}\}$, *the more compact yet exact result should be* $[0,12]$. *The constraint* $\mathcal{C}_x$ *is lost along the operations.*

2.2.1.3 Wrapping effect

The envelope effect is used to describe the pessimism caused by the way an interval vector (or matrix) is represented. One can always use an example in [Moore 1966] to explain it.

Example 2.4 *Consider successive rotations of a dimension 2 box* $\boldsymbol{x} = \boldsymbol{x}_1 \times \boldsymbol{x}_2$ *submitted to a rotation matrix:*

$$M = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}.$$

The first rotation of the box gives us a rectangle of same size, but different angle w.r.t the original axes. The representation of this rectangle by a box, generates a new rectangle enveloping the exact result. The borders of new rectangle are parallel to the reference axes. Its size is larger than the point-wise result. The successive rotations make the result larger every time a new rectangle envelopes the point-wise result. Rotation in interval arithmetic is then a very conservative operation. The result is illustrated in Figure 2.1 where $\boldsymbol{x}(0)$ is the initial box, and $\boldsymbol{f}(x)$ is a natural inclusion function which is explained in a further section.

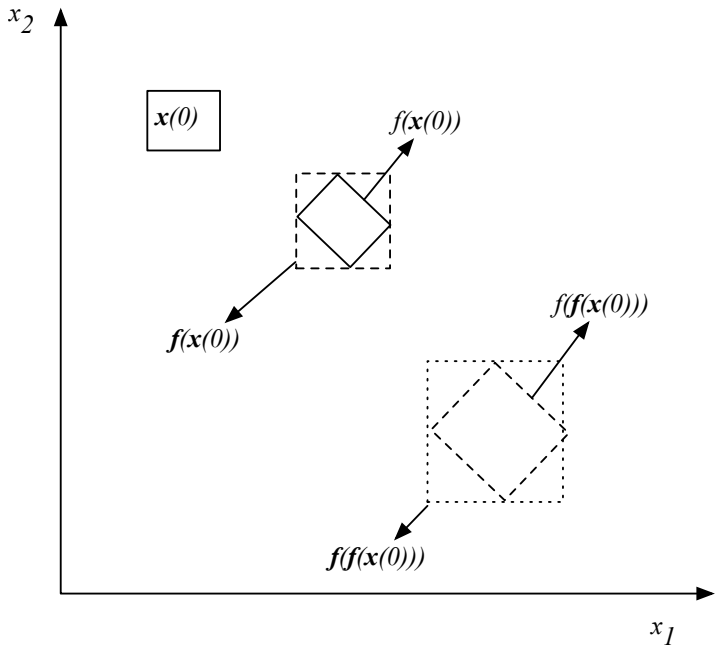

Figure 2.1: Rotation operation of an interval vector and the wrapping effect.

2.2.1.4 Modal arithmetic

The envelope evaluated from interval arithmetic can be over-bounded or under-bounded, as shown in Figure 2.2. The pessimism of interval arithmetic generate over-bounded envelope. To reduce this problem, other methods and arithmetic have been proposed. These

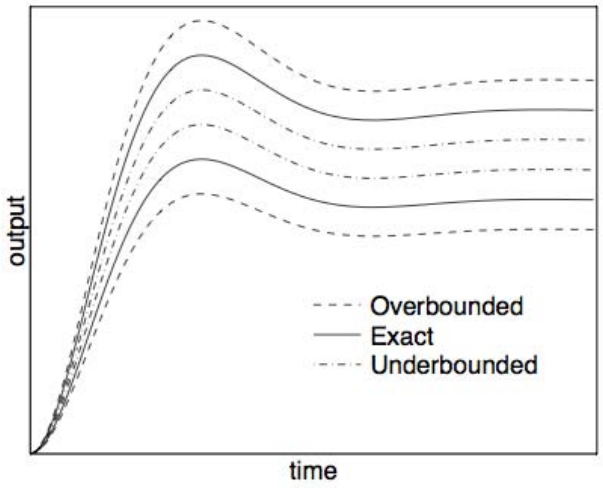

Figure 2.2: Evaluated envelope from interval arithmetic properties

researches remain most of time in mathematical fields. Among them, modal interval analysis is widely known and has been applied on real applications.

One of the consequences of the dependence effect is that there exists no zero point by subtraction of two identical interval numbers (except for degenerated intervals). In such case, if one needs to solve an interval equation with unknown variable, the obtained value could not hold the equality of original equation. An example can help to present the problem:

Example 2.5 *Consider a linear equation with unknown variable* $\boldsymbol{x}$*,*

$$[2,3] + \boldsymbol{x} = [3,6],$$

it can be solved with the natural interval arithmetic, which gives:

$$\boldsymbol{x} = [3,6] - [2,3] = [0,4].$$

But when $\boldsymbol{x} = [0,4]$ *is injected into the equation, equality does not hold but only inclusion:*

$$[2,3] + [0,4] = [2,7] \supset [3,6].$$

As a matter of fact, natural interval arithmetic encloses the solution set ($[3,6] \subset [2,7]$*), but it is fairly conservative.*

Modal interval analysis ([Gardeñes 1986]) was proposed to solve this problem. It is an algebraic and semantic extension of classical interval analysis. The essential idea is to introduce the concept of *proper/improper*. A modal interval is sometimes called a generalized interval. The set of generalized intervals is denoted by $\mathbb{KR}$ and is divided into three subsets: $\boldsymbol{x} := [\underline{\boldsymbol{x}}, \overline{\boldsymbol{x}}] \in \mathbb{KR}$ is called *proper* when $\underline{\boldsymbol{x}} \leq \overline{\boldsymbol{x}}$ and *improper* when $\underline{\boldsymbol{x}} \geq \overline{\boldsymbol{x}}$, and the *degenerated* when $\underline{\boldsymbol{x}} = \overline{\boldsymbol{x}}$. Then it is suggested to use $\mathbb{IR} = [\underline{\boldsymbol{x}}, \overline{\boldsymbol{x}}] \mid \underline{\boldsymbol{x}} \leq \overline{\boldsymbol{x}}$ to represent the set of

proper intervals, and $\overline{\mathbb{IR}} = [\underline{x}, \overline{x}] \mid \underline{x} \geq \overline{x}$ to represent the set of improper intervals. The acceptance of left bound superior to the right bound changes several basic definitions, such as $w(\boldsymbol{x}) := |\overline{x} - \underline{x}|$. A real function $f(x)$, where $x \in \mathbb{R}^n$, can be extended to $f(\boldsymbol{x})$ where $\boldsymbol{x} \in \mathbb{KR}^n$, which is called *modal extension*; in some works it can vary to $\mathbb{KR}$-extension or AE-extension ([Popova 1998]). The real arithmetic is extended to the so-called Kaucher arithmetic ([Goldsztejn 2006]). To manipulate modal intervals, three devoted operators are needed, which are *pro*, *imp* and *dual* defined in Kaucher arithmetic. For a modal interval $\boldsymbol{x} = [\underline{x}, \overline{x}] \in \mathbb{KR}$, we have:

$$pro\boldsymbol{x} = [min(\underline{x}, \overline{x}), max(\underline{x}, \overline{x})], imp\boldsymbol{x} = [max(\underline{x}, \overline{x}), min(\underline{x}, \overline{x})],$$

particularly there is

$$dual[\underline{x}, \overline{x}] := [\overline{x}, \underline{x}].$$

These operators build a relationship between proper and improper intervals. The other arithmetic operations stay same.

The set of the modal intervals is closed under the Kaucher arithmetic operations. A generalized interval $\boldsymbol{x}_1$ is an algebraic solution of the equation $f(\boldsymbol{x}) = \boldsymbol{b}$, where $\boldsymbol{x}$ is unknown, if the original algebraic relation is still valid when the variable $\boldsymbol{x}$ is replaced by the interval result $\boldsymbol{a}$. This is called algebraic closure. Take again the previous example:

Example 2.6 *Given a linear equation with unknown variable* $\boldsymbol{x}$*,*

$$[2, 3] + \boldsymbol{x} = [3, 6],$$

now it can be solved by modal interval arithmetic, which gives:

$$\boldsymbol{x} = [3, 6] - dual[2, 3] = [3, 6] - [3, 2] = [1, 3].$$

if we inject $\boldsymbol{x} = [1, 3]$ *to original place of first equation, the algebraic equality remains:*

$$[2, 3] + [1, 3] = [3, 6] = [3, 6].$$

Another characterization of generalized intervals is the modal semantic extension. Unlike classical interval analysis which identifies an interval by a set of real numbers only, modal interval arithmetic identifies an interval by a set of predicates which is fulfilled by real numbers. Each interval $\boldsymbol{x} \in \mathbb{KR}$ has an associated logical quantifier, either existential ($\exists$) or universal ($\forall$). For a real relation $f(x) = z$ where $x \in \mathbb{R}^n$ and $z \in \mathbb{R}$, the semantics of its modal extension can be expressed with the quantifiers, which are derived based on the modalities of generalized intervals. As universal quantifiers precede existential ones, such quantified propositions have the form of:

$$(\forall x_P \in \boldsymbol{x}_P)(Q_z z \in pro(z))(\exists x_I \in pro(\boldsymbol{x}_I))(f(x) = z),$$

where P and I are disjoint sets of index for proper and improper components of $\boldsymbol{x}_{P \cup I} \in \mathbb{KR}^n$,$Q_z = \forall$ if $\boldsymbol{z} \in \overline{\mathbb{IR}}$,$Q_z = \exists$ if $\boldsymbol{z} \in \mathbb{IR}$. For example:

Example 2.7

$$[1,3]+[4,5]=[5,8],$$
$$[1,3]+[5,4]=[6,7],$$
$$[3,1]+[5,4]=[8,5],$$
$$[3,1]+[4,5]=[7,6].$$

are all valid in modal interval arithmetic, they are interpreted as:

$$(\forall a\in[1,3])(\forall b\in[4,5])(\exists c\in[5,8])(a+b=c),$$
$$(\forall a\in[1,3])(\exists b\in[4,5])(\exists c\in[6,7])(a+b=c),$$
$$(\exists a\in[1,3])(\exists b\in[4,5])(\forall c\in[5,8])(a+b=c),$$
$$(\exists a\in[1,3])(\forall b\in[4,5])(\forall c\in[5,8])(a+b=c).$$

Though the modal interval arithmetic could give a perfect semantic for the zero point intervals and maintains the algebraic equality and logical explanation, the whole idea is still not yet adopted widely and sometimes complicated to interpreter. Most development tools, like *INTLAB* are still based on natural interval analysis, in which the inferior upper bound is not allowed. There is no point to build up the work from ground zero with this proposition, nevertheless it shows clearly the various potential of interval analysis in the future.

2.2.2 Inclusion function

Once the basic notions of interval analysis are defined, it is possible to evaluate vectorial functions with interval variables and finite number of arithmetic elementary operations. Different evaluation techniques have different efficiency or result.

The interval function $\boldsymbol{f}$ of $\mathbb{IR}^n$ in $\mathbb{IR}$ is called inclusion function of f if and only if:

$$\forall \boldsymbol{x}\in\mathbb{IR}^n, f(\boldsymbol{x})\subseteq \boldsymbol{f}(\boldsymbol{x}). \tag{2.9}$$

It is also important to notice that if the inclusion function of each component f_i of $\boldsymbol{f}$ satisfies:

$$\boldsymbol{f}_i(\boldsymbol{x})\supseteq f_i(\boldsymbol{x})(i=1,...,m),$$

we have:

$$\boldsymbol{f}=\boldsymbol{f}_1(\boldsymbol{x})\times...\times\boldsymbol{f}_m(\boldsymbol{x}). \tag{2.10}$$

One of the major objectives is to find the smallest box containing $f(\boldsymbol{x})$ for all $\boldsymbol{x}\in\mathbb{IR}^n$, called the minimal inclusion function for f, noted by $\boldsymbol{f}^*$.

$$\forall \boldsymbol{x}\in\mathbb{IR}^n, \boldsymbol{f}^*(\boldsymbol{x})=f(\boldsymbol{x}). \tag{2.11}$$

Definition 2.10 *(Convergent inclusion function) An inclusion function is convergent if and only if for any interval boxes $\boldsymbol{x}_k$, we have:*

$$\lim_{k\to\infty} w(\boldsymbol{x}_k) = 0 \Rightarrow \lim_{k\to\infty} w(\boldsymbol{f}(\boldsymbol{x}_k)) = 0, \tag{2.12}$$

which implies,

$$\forall \boldsymbol{x} \in \mathbb{IR}^n, \boldsymbol{f}(\boldsymbol{x}) = f(\boldsymbol{x}). \tag{2.13}$$

Definition 2.11 *(Monotonic inclusion function) An inclusion function is monotone if and only if:*

$$\boldsymbol{x} \subset \boldsymbol{y} \rightarrow \boldsymbol{f}(\boldsymbol{x}) \subset \boldsymbol{f}(\boldsymbol{y}). \tag{2.14}$$

with $\boldsymbol{x}$ and $\boldsymbol{y}$ two boxes of $\mathbb{IR}^n$.

2.2.2.1 Elementary inclusion function

Firstly, the elementary functions should be evaluated based on the properties introduced in previous sections. Generally elementary real functions can be extended to interval space with following formula:

$$f(\boldsymbol{x}) = [\underset{x\in\boldsymbol{x}}{inf}(f(x)), \underset{x\in\boldsymbol{x}}{sup}(f(x))], \tag{2.15}$$

where $inf(\bullet)$ and $sup(\bullet)$ are two operators returning inferior and superior bounds respectively, as defined for interval variables in previous sections.

It is then easy to write down the expressions of different monotone elementary functions, for example:

$$\begin{aligned}
exp(\boldsymbol{x}) &= [exp(\underline{\boldsymbol{x}}), exp(\overline{\boldsymbol{x}})], \\
ln(\boldsymbol{x}) &= \begin{cases} [-\infty, ln(\overline{\boldsymbol{x}})] \; if \; 0 \in \boldsymbol{x}, \\ \emptyset \; if \; \overline{\boldsymbol{x}} \leqslant 0, \\ [ln(\underline{\boldsymbol{x}}), ln(\overline{\boldsymbol{x}})] \; otherwise, \end{cases} \\
\boldsymbol{x}^2 &= \begin{cases} [0, max(\underline{\boldsymbol{x}}^2, \overline{\boldsymbol{x}}^2)] \; if \; 0 \in \boldsymbol{x}, \\ [min(\underline{\boldsymbol{x}}^2, \overline{\boldsymbol{x}}^2), max(\underline{\boldsymbol{x}}^2, \overline{\boldsymbol{x}}^2)] \; otherwise, \end{cases} \\
\sqrt{\boldsymbol{x}} &= \begin{cases} [-\sqrt{\overline{\boldsymbol{x}}}, -\sqrt{\underline{\boldsymbol{x}}}] \; et \; [\sqrt{\underline{\boldsymbol{x}}}, \sqrt{\overline{\boldsymbol{x}}}] \; if \; \underline{\boldsymbol{x}} \geqslant 0, \\ [0, \sqrt{\overline{\boldsymbol{x}}}] \; if \; not. \end{cases}
\end{aligned} \tag{2.16}$$

Other non monotone functions need specific analysis and algorithm.

Example 2.8

$$\begin{aligned} exp([0,1]) &= [1,e], \\ log([-2,-1]) &= \emptyset, \\ [-2,2]^2 &= [0,4], \\ [-2,2] \times [-2,2] &= [-4,4], \\ \sqrt{[-2,4]} &= [0,2], \\ abs([-2,1]) &= [0,2], \\ sin([0,\frac{2\pi}{3}]) &= [0,1], \\ cos([0,\frac{2\pi}{3}]) &= [-\frac{1}{2},1]. \end{aligned} \tag{2.17}$$

Let us notice that $[-2,2]^2$ and $[-2,2] \times [-2,2]$ provide different results, which is discussed in the following section.

2.2.2.2 Natural inclusion function

As it is important to find a convergent inclusion function for a given function $\boldsymbol{f}$, there exist various methods in different communities. The simplest and the most direct way consists in replacing every point-wise number by an interval in which it belongs, and also the real elementary functions by their interval extensions. This method provides the *"natural inclusion function"*.

Example 2.9 *Consider the function:*

$$f(x) = x^2 + 2x + 1.$$

A natural inclusion function of this function is:

$$\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{x}^2 + 2 \times \boldsymbol{x} + 1.$$

The evaluation of this function for $\boldsymbol{x} = [-1,1]$ *is:*

$$\boldsymbol{f}([-1,1]) = [-1,1]^2 + 2 \times [-1,1] + 1 = [-1,1] + [-1,3] = [-2,4].$$

Notice that the interval image of $\boldsymbol{x}$ *by* f *is* $f([-1,1]) = [0,4]$*, which verifies the inclusion relation* (2.9).

Lots of works ([Jaulin 2001c, Ratschek 1985]) have proven that natural inclusion functions are rarely minimal. The pessimism is always introduced by the fact that each occurrence of an interval variable is considered as an independent variable which respect to other occurrences of the same variable. A given function generally has different formulations, in natural interval arithmetic.

Example 2.10 *Consider the function $f : x \to x^2 + 2x + 2$. This function can be rewritten under the following forms:*

$$f_1(x) = x(x+2) + 2,$$
$$f_2(x) = x \times x + 2 \times x + 2,$$
$$f_3(x) = x^2 + 2 \times x + 2,$$
$$f_4(x) = (x+1)^2 + 1.$$

and for [x]=[-2,1], we have:

$$\boldsymbol{f}_1(\boldsymbol{x}) = \boldsymbol{x}(\boldsymbol{x}+2) + 2 = [-4, 5],$$
$$\boldsymbol{f}_2(\boldsymbol{x}) = \boldsymbol{x} \times \boldsymbol{x} + 2 \times \boldsymbol{x} + 2 = [-4, 8],$$
$$\boldsymbol{f}_3(\boldsymbol{x}) = \boldsymbol{x}^2 + 2 \times \boldsymbol{x} + 2 = [-2, 8],$$
$$\boldsymbol{f}_4(\boldsymbol{x}) = (\boldsymbol{x}+1)^2 + 1 = [1, 5].$$

It is clear that different expressions give different results, though the real interval image is included in all the results. Especially the expressions $\boldsymbol{x} \times \boldsymbol{x}$ and $\boldsymbol{x}^2$ do not give that same interval result, the interval variable $\boldsymbol{x}$ is supposed to take the same value in any occurrence during the calculation, but in first expression it appears two times and it is considered as different values. Some constraints are hence lost and pessimism is introduced during the evaluation of the function f. $\boldsymbol{f}_4$ gives the smallest interval solution, and it can be proven to be minimal.

The first observation on pessimism of interval arithmetic is that the width of obtained interval results depends on the number of occurrences of each interval variable in the expression of function f.

To reduce the pessimism introduced by multi-occurrences, one should always find the equivalent expression which reduces the number of occurrences of each variable. If each variable appears just once in function f, the natural inclusion function is minimal. Unfortunately there is no method which can be used systematically to find the minimal inclusion function. The natural inclusion function is still the simplest way to evaluate an interval function.

2.2.2.3 Centered inclusion function

One of the methods that can reduce the pessimism is the Taylor series function which is developed based on the arithmetic of Taylor form [Neumaier 2003]. It constitutes a significant enhancement of the toolkit of interval analysis techniques. The use of the Taylor expansion function as a wrapping effect controller can be found in [Ramdani 2010].

If the interval $\boldsymbol{x}$ is not too wide, we can use the centered form of the inclusion function. Given a function $\boldsymbol{f} : \mathbb{R}^n \to \mathbb{R}^m$ differentiable and to be derived continuously on the domain $D \subset \mathbb{R}^n$, then:

$$\forall x \in D, f(x) \in \boldsymbol{f}_m(\boldsymbol{x}) = \boldsymbol{f}(m) + \boldsymbol{f}'(\boldsymbol{x})(\boldsymbol{x} - m), \tag{2.18}$$

where m is a point in the interval $\boldsymbol{x}$ (generally the center) and $\boldsymbol{f}'(\boldsymbol{x})$ is the inclusion Jacobian function of f, $\boldsymbol{f}_m$ is the centered inclusion function. Notice that in the interval vector case, the center of $\boldsymbol{x}$ is written m, the i_{th} element of $\boldsymbol{x}$ has the center of m_i.

The centered inclusion function gives generally a less pessimistic result compared to the natural inclusion function when $w[x]$ is not too large or when the system equations are complex on differential. Actual systems are often described by ordinary differential equations. Interval analysis and a first-order enclosure of the solution to the ordinary differential equation allow one to compute guaranteed solutions of the state estimation problem. Then, validated numerical methods for solving the ordinary differential equation are applied. These methods use high-order interval Taylor models ([Corliss 1994]) to compute intervals which are guaranteed to contain the solution of the ordinary differential equation.

2.2.2.4 Convergence of the inclusion function

The criterion of the convergence of inclusion functions is studied by Moore [Moore 1979]. In his work, the convergence order of a inclusion function is defined as the largest integer α which satisfies:

$$\exists \beta \in \mathbb{R}^+ | \left(w(\boldsymbol{f}(\boldsymbol{x})) - w(f(\boldsymbol{x}))\right) \leq \beta w(\boldsymbol{x})^{\alpha}. \tag{2.19}$$

The convergence order of a minimal inclusion function is infinite. It shows that the center inclusion function is more interesting than the natural inclusion function when the width of intervals is small. But for larger size intervals, it is preferable to use natural inclusion function.

2.3 Tighten interval calculation

Two main methods have been proposed during the last decades to solve equation systems. One is known as the *set inversion problem*, the other as rooted in *constraint satisfaction*.

2.3.1 Set inversion problem

Considering the following equation:

$$f(x) = y,$$

the objective is to find:

$$f(x) \in \boldsymbol{y}, x \in \boldsymbol{x}. \tag{2.20}$$

$\boldsymbol{x}$ is the set of values of x which satisfy the equation, and $\boldsymbol{y}$ is an interval vector. The solution set $\mathbb{S}$ of such equation is given by:

$$\mathbb{S} = \{x \in \boldsymbol{x} \mid f(x) \in \boldsymbol{y}\}, \tag{2.21}$$

which is equivalent to:

$$\mathbb{S} = f^{-1}(\boldsymbol{y}) \cap \boldsymbol{x}. \tag{2.22}$$

Firstly the set $\mathbb{S}$ defined in the previous equation is the "set inversion problem". For linear systems, one can use natural arithmetic to solve the equation, but results may be pessimist; for nonlinear systems, the performance of any method based on centered forms is drastically affected by pessimisms [Uwamusi 2008].

This problem can be partially solved with an approximate but also guaranteed way based on interval analysis and an algorithm called SIVIA (*Set Inversion Via Interval Analysis*) which was developed by [Jaulin 1993]. This algorithm enables the construction of two sub-paving $\underline{\mathbb{S}}$ and $\overline{\mathbb{S}}$ which verify the following inclusions:

$$\underline{\mathbb{S}} \subset \mathbb{S} \subset \overline{\mathbb{S}}. \tag{2.23}$$

The algorithm SIVIA allows one to obtain these two sub-pavings with a required precision ε, based on an inclusion test. The relation between the two sub-pavings can be characterized as:

$$\Delta\mathbb{S} = \overline{\mathbb{S}} - \underline{\mathbb{S}}. \tag{2.24}$$

where $\Delta\mathbb{S}$ is called the inclusion test uncertainty, in which no decision can be made during the test. The properties of solutions are:

- if $\overline{\mathbb{S}} = \emptyset$ the problem (2.20) has no solution,
- if $\underline{\mathbb{S}} \neq \emptyset$, there exists at least one verified solution for (2.20).

2.3.1.1 Inclusion test

An inclusion test aims at verifying whether an interval, which is calculated with an inclusion function, belongs to a *a priori* known set. For any interval box $\boldsymbol{x}$, three situations are anticipated:

- $\boldsymbol{f}(\boldsymbol{x}) \subseteq \boldsymbol{y} \rightarrow \boldsymbol{x} \subset \mathbb{S}$ and $\boldsymbol{x}$ is called *feasible*, or *acceptable*,
- $\boldsymbol{f}(\boldsymbol{x}) \cap \boldsymbol{y} = \emptyset \rightarrow \mathbb{S} - \boldsymbol{x}$ and $\boldsymbol{x}$ is called *unfeasible*, or *rejectable*,
- $\boldsymbol{f}(\boldsymbol{x}) \cap \boldsymbol{y} \neq \emptyset \rightarrow \boldsymbol{x}$ is called *uncertain*.

One of the purposes of SIVIA is to deal with the last situation, which is the method of bisection.

2.3.1.2 SIVIA algorithm

The SIVIA algorithm determines the sub-paving $\underline{\mathbb{S}}$ which contains the solutions and $\overline{\mathbb{S}}$ which contains the admissible boxes and undetermined boxes. This algorithm uses stacks: a stack $\mathcal{P}_{int}$ keeps all the intervals to-be-analyzed (the intermediate boxes). An initial box $\boldsymbol{x}(0) \in \mathbb{X}_0$ is supposed to contain all the solutions and it is placed in the stack $\mathcal{P}_{int}$.

During the execution of the algorithm, the first element of such stack is retrieved as a box $\boldsymbol{x}$. This operation extracts the element on the top of the stack. Its image by the inclusion function $\boldsymbol{f}$ is then tested against the known set $\boldsymbol{y}$. If the box $\boldsymbol{x}$ is undetermined and its length is superior to ε (a user predefined size), it is then bisected. The two sub-pavings $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are put on the top of the stack $\mathcal{P}_{int}$. If the box $\boldsymbol{x}$ is acceptable, $\boldsymbol{x}$ is added to the stack $\underline{\mathbb{S}}$ and $\overline{\mathbb{S}}$, or it is undetermined but it can no longer be divided, $\boldsymbol{x}$ is added to the stack $\overline{\mathbb{S}}$

Algorithm 1 Algorithm SIVIA($\boldsymbol{f}$,$\boldsymbol{y}$,$\mathcal{P}_{int}$,ε)

Input: $\boldsymbol{f}$,$\boldsymbol{x}(0)$,$\boldsymbol{y}$,ε;
Output: $\underline{\mathbb{S}}$, $\overline{\mathbb{S}}$;

initialization: $\mathcal{P}_{int} := \boldsymbol{x}(0)$;
$\boldsymbol{x}$:=$getTop(\mathcal{P}_{int})$;
if $\boldsymbol{f}(\boldsymbol{x}) \subset \boldsymbol{y}$ **then**
  $\mathcal{P}_{ad} := \mathcal{P}_{ad} \cup \boldsymbol{x}$;
  $\underline{\mathbb{S}} := \underline{\mathbb{S}} \cup \boldsymbol{x}$; $\overline{\mathbb{S}} := \overline{\mathbb{S}} \cup \boldsymbol{x}$;
else if $\boldsymbol{f}(\boldsymbol{x}) \cap \boldsymbol{y} \neq \emptyset$ AND $w(\boldsymbol{x}) < \varepsilon$ **then**
  $\overline{\mathbb{S}} := \overline{\mathbb{S}} \cup \boldsymbol{x}$;
else if $\boldsymbol{f}(\boldsymbol{x}) \cap \boldsymbol{y} \neq \emptyset$ AND $w(\boldsymbol{x}) > \varepsilon$ **then**
  $bisectBox(\boldsymbol{x}) \rightarrow \{\boldsymbol{x}_1, \boldsymbol{x}_2 | \boldsymbol{x}_1 \cup \boldsymbol{x}_2 = \boldsymbol{x}\}$;
  $\mathcal{P}_{int} := \mathcal{P}_{int} \cup \boldsymbol{x}_1$, $\mathcal{P}_{int} := \mathcal{P}_{int} \cup \boldsymbol{x}_2$;
end if
if $\mathcal{P}_{int} \neq \emptyset$ **then**
  SIVIA($\boldsymbol{f}$,$\boldsymbol{y}$,$\mathcal{P}_{int}$,ε);
end if

The function $getTop(\circ)$ retrieves a box from the stack, and $bisectBox(\circ)$ divide box into two sub-box. The box stack $\Delta\mathbb{S} = \overline{\mathbb{S}} - \underline{\mathbb{S}}$ which represents the uncertainty of the solution set contains the undetermined boxes with dimension smaller than predefined by ε. The volume of this set reduces while ε gets smaller.

It is clear that SIVIA is a recursive algorithm, its complexity is exponential depending on the size of the variable vector. The number of bisections is estimated as inferior to:

$$\left(\frac{w(\boldsymbol{x}(0))}{\varepsilon} + 1\right)^n. \tag{2.25}$$

$[\boldsymbol{x}](0)$ is the initial search box as mentioned above, n is the dimension of vector $\boldsymbol{x}$. This number can be reduced by using contraction or preconditioning methods, which are discussed later.

Example 2.11 *Consider*

$$\boldsymbol{y} = \boldsymbol{x}_1^4 - \boldsymbol{x}_1^2 + 4\boldsymbol{x}_2^2,$$

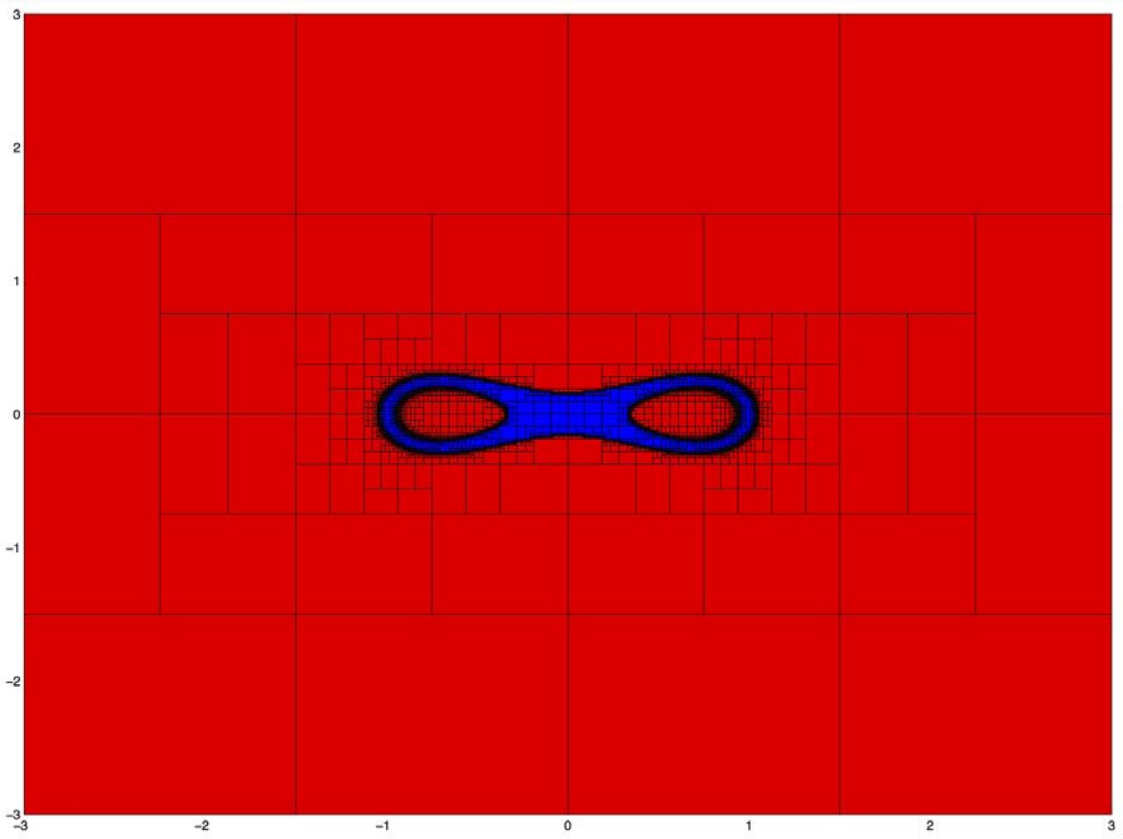

Figure 2.3: SIVIA solution illustration

where $\boldsymbol{y} = [-0.1, 0.1]$, *and* $\boldsymbol{x}(0) = ([-3, 3][-3, 3])^T$. *To find the solution set S, SIVIA can be used and the result is shown in the Figure 2.3.*

The strategy of bisection is an important issue, which can influence the efficiency of the algorithm. An overview of different strategies is given in the next section.

2.3.1.3 Bisection strategies

SIVIA uses a branch-and-bound type algorithm to explore and refine the research space $\boldsymbol{x}(0)$. Several works have proposed different strategies to carry out bisection of the search space subject [Ratz 1995, Csendes 1997, Csendes 2003, Csendes 2004]. As the bisection rules have been implemented in different applications, it is worth mentioning them briefly here.

Consider a general set inversion problem:

$$\mathbb{S} = \{\boldsymbol{x} \in \mathbb{X} \mid \boldsymbol{f}(\boldsymbol{x}) \in \boldsymbol{y}\},$$

exactly as in the previous section, which can be solved by the algorithm SIVIA [Jaulin 1993]. The bisection strategy consists in choosing the component of the interval vector to bisect.

Mostly it is based on a dedicated function $D(i)$. The main target is to find an optimal component also called coordinate direction to bisect the box. This choice is named interval subdivision direction selection. It is generalized as:

$$k = arg\left(\max_{i\in(1,\dots,n)}(D(i))\right), \boldsymbol{x} \in \mathbb{IR}^n. \tag{2.26}$$

Strategy A This rule is based on the interval component width. Based on the definition 2.6, in the Cartesian product the bisection direction choice is given by the largest vector component width. Another interpretation can be given as:

$$D(i) = w(\boldsymbol{x}_i). \tag{2.27}$$

Strategy B In [Walster 2004] Walter and Hansen proposed a bisection rule which aims to maximize the quantity:

$$q_i = \left\{\max_{t\in\boldsymbol{x}_i}\Big(f\big(m(\boldsymbol{x}_1),\dots,m(\boldsymbol{x}_{i-1}),t,m(\boldsymbol{x}_{i+1})\big)\Big) - \min_{t\in\boldsymbol{x}_i}\Big(f\big(m(\boldsymbol{x}_1),\dots,m(\boldsymbol{x}_{i-1}),t,m(\boldsymbol{x}_{i+1})\big)\Big)\right\}. \tag{2.28}$$

The quantity q_i gives an indication of the variation of the function when t slides in $\boldsymbol{x}_i$. When the function is not monotonic, the process is very complex, it is suggested to use the following function:

$$D(i) = w(\boldsymbol{f}'_i(\boldsymbol{x})) \cdot w(\boldsymbol{x}_i). \tag{2.29}$$

in which $\boldsymbol{f}'_i$ is the i_{th} element of the inclusion function of the gradient function of f.

Strategy C This strategy is interesting as it aims at finding a bisection direction which can minimize the size of $\boldsymbol{f}(\boldsymbol{x})$ by using a centered inclusion function. It is based on the Equation (2.18), the size of the image is given by:

$$\begin{aligned} w(\boldsymbol{f}_m(\boldsymbol{x})) &= w(\boldsymbol{f}(m) + (\boldsymbol{f}(\boldsymbol{x}))(\boldsymbol{x} - m)) \\ &= \sum_{i=1}^{n} w((\boldsymbol{f}_i(\boldsymbol{x}))(\boldsymbol{x}_i - m_i)). \end{aligned} \tag{2.30}$$

The bisection is chosen according to the following cost function:

$$D(i) = w((\boldsymbol{f}_i(\boldsymbol{x}))(\boldsymbol{x}_i - m_i)). \tag{2.31}$$

Strategy D It is a variation of strategy A, for the cases where the difference of the absolute width values of each component are big so that the bisection direction would be stuck in one component. We can then use an analogue expression but with percentage value of actual width to the original width of each component.

$$D(i) = \frac{w(\boldsymbol{x}_i)}{w(\boldsymbol{x}_i(0))}, \tag{2.32}$$

where $w(\boldsymbol{x}_i(0)$ is the original width of the i_{th} interval component.

The choice of strategy is based on the algorithm requirements, considering a balance between efficiency, convergence, implementation complexity and algorithm speed.

2.3.2 Constraint satisfaction problem

Another approach of solving an interval equation system is to use constraint propagation [Jaulin 2001b]. As a matter of fact, the inclusion relations and the equations can be interpreted as the constraints; the resolution of such system can then be taken into a Constrain Satisfaction Problem (CSP). Let us recall the basic definitions:

Definition 2.12 *A Constraint Satisfaction Problem $H = (\mathcal{X}, \mathcal{D}, \mathcal{C})$ is defined by:*
- a set of variables $\mathcal{X} = \{x_1, ..., x_n\}$,
- a set of value domains $\mathcal{D} = \{D_1, ..., D_n\}$ where D_i is the domain associated to the variable x_i,
- a set of constraints $\mathcal{C} = \{C_1, ..., C_m\}$, linking the variables $\mathcal{X}$.

For example, an interval linear system of the form

$$0 \in \boldsymbol{AX} - \boldsymbol{B},$$

can be represented as a CSP as follows:

$$CSP(A \in \boldsymbol{A}, B \in \boldsymbol{B}, X \in \boldsymbol{X}, AX = B).$$

We can also write the CSP row by row:

$$H = \begin{pmatrix} \mathcal{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\} \\ \mathcal{D} = \{\mathbb{IR}, ..., \mathbb{IR}\} \\ \mathcal{C} = \left\{ \mathcal{C}_i : 0 \in \sum_{k=1}^{n} \boldsymbol{a}_{i,k}\boldsymbol{x}_k - \boldsymbol{b}_i \right\} \end{pmatrix}. \tag{2.33}$$

The solution S of the $CSP : H = (\mathcal{X}, \mathcal{D}, \mathcal{C})$ is the set of all the values affected to the corresponded variables at the same time.

2.3.2.1 Contraction and consistency of a CSP

The resolution of CSP starts from an infinite domain or a bounded domain. The reduction of the domain is known as a local consistency problem, which can take the form of *node consistency, arc consistency, and path consistency* [Chabert 2009, Lhomme 1993]. The operation is called *constraint propagation* or *contraction*, which is based on the equivalent relation below:

Definition 2.13 *Two CSP $\mathcal{H}_1$ and $\mathcal{H}_2$ are equivalent if and only if they have the same set of solutions.*

Remark 2.1 *For the same set of variables $\mathcal{X}$ and the same set of constraints $\mathcal{C}$, different sets of variable domains $\mathcal{D}_i$ define different CSP $\mathcal{H}_i$.*

Definition 2.14 *A contractor $\mathcal{R}$ for a CSP $\mathcal{H}_1 = (\mathcal{X}, \mathcal{D}_1, \mathcal{C})$ is an operator that can shrink the defined domain $\mathcal{D}_1$ into a domain $\mathcal{D}_2$ without losing any solution, such that:*

$$\mathcal{D}_2 \subset \mathcal{D}_1.$$

The new CSP $\mathcal{H}_2$ is equivalent to $\mathcal{H}_1$.

A *CSP* is solvable when it is equivalent to the *CSP* in which the infinite quantity domain is replaced by a largest value in serving computer (CSPs are solved by the help of modern computer science). The contractor aims to reduce the initial domain into an as small as possible domain. The principle is to reject the parts of the domain which are not consistent with the constraints.

Definition 2.15 *A CSP: $\mathcal{H} = (\mathcal{X}, \mathcal{D}, \mathcal{C})$ is globally consistent if and only if:*

$$\forall x_i \in D_i, \exists (x_1, ..., x_i, ..., x_n) \in \mathcal{D} \mid \forall C(x_1, ..., x_i, ..., x_n) \in \mathcal{C}, C(x_1, ..., x_i, ..., x_n) \text{ is verified},$$

in which $C(x_1, ..., x_i, ..., x_n)$ is a single constraint with a set of varibles.

Definition 2.16 *The number of constraints of the CSP $\mathcal{H} = (\mathcal{X}, \mathcal{D}, \mathcal{H})$ is noted N_c. The number of the variables of CSP $\mathcal{H} = (\mathcal{X}, \mathcal{D}, \mathcal{H})$ is noted N_x;*

Global consistency can be interpreted as for all the variables in the problem, the defined domain corresponds to the variation of the constraints. In such case, a globally consistent *CSP* gives a minimal exterior estimation of the equivalent system equation of the solution.

There is a large choice of contractors. Each has its own advantages and shortcomings, system characteristics and available information. We use these criteria to classify different contractors: constraint linearity, number of unknown variables, constraints, and size of $\boldsymbol{X}$, which is $w(\boldsymbol{X})$. The first criterion to consider is the linearity of the constraints, which define two categories, linear *CSP*s and non linear *CSP*s.

2.3.2.2 Convergence of contractors

The contraction is an operation that can reduce the search space to a domain no longer compressible according to the same contractor. Its size depends on the initial uncertainty of the system. The result is yielded by an iterative process:

$$\mathcal{D}_{k+1} = \mathcal{R}(\mathcal{D}_k).$$

The algorithm stops is at the point when:

$$\mathcal{D}_{k+1} = \mathcal{R}(\mathcal{D}_k) = \mathcal{D}_k.$$

The solution $\mathcal{S}$ of *CSP* $\mathcal{H} = (\mathcal{X}, \mathcal{D}, \mathcal{C})$ can be provided by the suitable contractor for this *CSP* which verifies the properties:

$$\begin{aligned}&\text{contractible } \boldsymbol{X} \subset \mathcal{D} \Rightarrow \mathcal{R}(\boldsymbol{X}) \in \mathcal{D}\\&\text{monotony } \boldsymbol{X} \subset \mathcal{D}, \boldsymbol{X}' \subset \mathcal{D}; \boldsymbol{X} \subset \boldsymbol{X}' \rightarrow \mathcal{R}(\boldsymbol{X}) \subset \mathcal{R}(\boldsymbol{X}')\\&\text{idempotent } \boldsymbol{X} = \mathcal{S} \rightarrow \mathcal{R}(\boldsymbol{X}) = \boldsymbol{X}\end{aligned}$$

Monotonicity is a property of the inclusion function. If the the series $\mathcal{D}_{k+1} = \mathcal{D}_k \cap \mathcal{R}(\mathcal{D}_k)$ is an inclusion function, it should be monotonic which yields to a minimal function. Among all the contractors, the fix point contractors are those that are idempotent.

In this thesis, the solution of a *CSP* is noted as $\mathcal{D}_s$. For convergent *CSP*, theoretically we have:

$$\mathcal{D}_s = \mathcal{D}_{k+1} = \mathcal{R}(\mathcal{D}_k) = \mathcal{D}_k.$$

In practice, the convergence is validated when the different of two domains from two successive iterations are less than a predefined size.

2.3.2.3 Linear *CSP*

These *CSP*s concern linear systems, which can be represented by following equation:

$$Ax - B = 0$$

Gauss Elimination Contractor ($\mathcal{C}_{GE}$) This contractor is for the *CSP* $\mathcal{H} : (\mathcal{X}, \mathcal{D}, \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} = 0)$.

The principle is to transform the interval matrix $\boldsymbol{A}$ into two triangular matrices, the inferior $\boldsymbol{L}$ and superior $\boldsymbol{U}$ such that $\boldsymbol{A} \subseteq \boldsymbol{L}\boldsymbol{U}$. This manipulation is also called LU decomposition.

$$\begin{pmatrix} 1 & 0 & \cdots & 0 \\ \boldsymbol{l}_{2,1} & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{l}_{n,1} & \boldsymbol{l}_{n,2} & \cdots & 1 \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_{1,1} & \boldsymbol{u}_{2,1} & \cdots & \boldsymbol{u}_{1,n} \\ 0 & \boldsymbol{u}_{2,2} & \cdots & \boldsymbol{u}_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{u}_{n,n} \end{pmatrix} \supseteq \begin{pmatrix} \boldsymbol{a}_{1,1} & \boldsymbol{a}_{1,2} & \cdots & \boldsymbol{a}_{1,n} \\ \boldsymbol{l}_{2,1} & \boldsymbol{a}_{2,2} & \cdots & \boldsymbol{a}_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{a}_{n,1} & \boldsymbol{a}_{n,2} & \cdots & \boldsymbol{a}_{n,n} \end{pmatrix}$$

The solution is given in the following two CSPs:

$$\begin{aligned}\boldsymbol{L}y &\supseteq \boldsymbol{b},\\ \boldsymbol{U}x &\subseteq \boldsymbol{y}.\end{aligned}$$

The matrices $\boldsymbol{L}$ and $\boldsymbol{U}$ are obtained by iterative calculus:

$$\boldsymbol{l}_{i,k} = \frac{\boldsymbol{a}_{i,k} - \sum_{j>k} \boldsymbol{l}_{i,j}\boldsymbol{u}_{j,k}}{\boldsymbol{u}_{k,k}}, 0 \notin \boldsymbol{u}_{k,k}, \text{for } i > k,$$

and

$$\boldsymbol{u}_{i,k} = \boldsymbol{a}_{i,k} - \sum_{j<i} \boldsymbol{l}_{i,j}\boldsymbol{u}_{j,k}, \text{ for } i \leq k,$$

which yields

$$\boldsymbol{y}_i = \boldsymbol{b}_i - \sum_{j=1}^{i-1} \boldsymbol{l}_{i,j}\boldsymbol{y}_j, \; i = 1, ..., n,$$

and the contractor is completed by

$$\boldsymbol{x}_i = \frac{\boldsymbol{y}_i - \sum_{j=i+1}^{n} \boldsymbol{u}_{i,j}\boldsymbol{x}_j}{\boldsymbol{u}_{i,i}}, \; i = 1, ..., n.$$

The algorithm is then given by Algorithm 2:

Algorithm 2 Algorithm Gauss elimination contractor $\mathcal{C}_{GE}(\mathcal{X}, \mathcal{D}_{\boldsymbol{x}}, \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} = 0)$

Input: $\mathcal{D}_{\boldsymbol{x}}, \boldsymbol{A} \in \mathbb{IR}^{n\times n}, \boldsymbol{b} \in \mathbb{IR}^{n\times 1}$;
Output: $\boldsymbol{x}$;

```
for i = 1 to n − 1 do
    if 0 ∈ a_{i,i} then
        (x_1, ..., x_n) = IR^n; return;
    end if
    for j = i + 1 to n do
        a_j = a_{j,i} / a_{i,i}; b_j = b_j − a_j × b_i;
        for k = i + 1 to n do
            a_{j,k} = a_{j,k} − x_j × a_{i,k};
        end for
    end for
end for
for i = n to 1 do
    x_i = (b − Σ_{j=i+1}^{n} a_{i,j} × x_j) / a_{i,i}.
end for
```

This contractor requires that $0 \notin \boldsymbol{a}_{i,i}$, otherwise the contractor fails to contract. $\mathcal{C}_{GE}$ is efficient when the interval matrix $\boldsymbol{A}$ is close to the identity matrix.

Gauss-Seidel Contractor ($\mathcal{C}_{GS}$) This contractor is for the *CSP* $\mathcal{H} : (\mathcal{X}, \mathcal{D}, \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} = 0)$.

The principle is to develop row by row the interval model $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, where $\boldsymbol{A}$ is supposed to be a square matrix so that $N_x = N_c$.

The algorithm starts from decomposing the matrix $\boldsymbol{A}$, whose diagonal elements should not be zero into two sub matrices:

$$\boldsymbol{A} = \boldsymbol{\Lambda} + \boldsymbol{M}.$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix whose diagonal elements are identical to the $\boldsymbol{A}$. $\boldsymbol{M}$ is the matrix whose diagonal elements are zeros. The solution of $\boldsymbol{x}$ of the *CSP* is the fixed point of:

$$\psi(\boldsymbol{x}) = \boldsymbol{\Lambda}^{-1}(\boldsymbol{b} - \boldsymbol{M}\boldsymbol{x}).$$

which yields:

$$\mathcal{C}_{GS} : \boldsymbol{x}_{k+1} = \boldsymbol{x}_k \cap (\boldsymbol{\Lambda}^{-1}(\boldsymbol{b} - \boldsymbol{M}\boldsymbol{x})). \tag{2.34}$$

Algorithm 3 Algorithm Gauss-Seidel contractor $\mathcal{C}_{GS}(\mathcal{X}, \mathcal{D}_{\boldsymbol{x}}, \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} = 0)$

Input: $\mathcal{D}_{\boldsymbol{x}}, \boldsymbol{A}, \boldsymbol{b}$;
Output: $\boldsymbol{x}$;
1: $\boldsymbol{\Lambda} = diag(\boldsymbol{A})$;
2: $\boldsymbol{M} = \boldsymbol{A} - \boldsymbol{\Lambda}$;
3: $\boldsymbol{i} = \boldsymbol{\Lambda}^{-1}(\boldsymbol{b} - \boldsymbol{M}\boldsymbol{x})$;
4: $\boldsymbol{x} = \boldsymbol{x} \cap [i]$.

2.3.2.4 All system *CSP*

The contractors presented below are suitable for the systems in the form:

$$f(x) = 0.$$

Constraint propagation contractor ($\mathcal{C}_P$) The principle is to decompose the constraint equation $f(\boldsymbol{x_1}, ..., \boldsymbol{x_n}) = 0$ in a sequence of elementary operations like $\{+, -, \times, /\}$ yielding a list of primitive constraints ([Lhomme 1996]). For example, equation:

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + K_{k+1}(y_{k+1} - C\hat{x}_{k+1|k}).$$

can be decomposed into the set of following primitive constraints:

$$\begin{cases} a_1 = C\hat{x}_{k+1|k}, \\ a_2 = y_{k+1} - a_1, \\ a_3 = K_{k+1}a_2, \\ \hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + a_3. \end{cases}$$

The algorithm can be repeated until the convergence of the result is verified.

Constraint forward backward contractor ($\mathcal{C}_{FB}$) This contractor is for the *CSP* with the form of $\mathcal{H} : (\mathcal{X}, \mathcal{D}, f(\boldsymbol{x}) = 0)$.

The forward-backward contractor is based on constraint propagation. The basis of $\mathcal{C}_{FB}$ is also to decompose the original constraints into elementary constraints. The sequence of operations should be arranged carefully as it has an impact on the contraction result.

Krawczyk contractor ($\mathcal{C}_K$) This contractor is for the *CSP* $\mathcal{H} : (\mathcal{X}, \mathcal{D}, f(\boldsymbol{x}) = 0)$. $f(\bullet)$ should be differentiable, the number of constraints N_c should be equal to the dimension of $\boldsymbol{x}$ N_x, i.e. we have $N_x = N_c$.

M is supposed to be an invertible matrix, so we can construct a function:

$$f(x) = 0 \Leftrightarrow x - Mf(x) = x \Leftrightarrow \psi(x) = x - Mf(x). \tag{2.35}$$

Once the function is established, the point $\boldsymbol{x}$ which fulfills $f(\boldsymbol{x}) = 0$ is the fixed point of the series defined by $\boldsymbol{x}_{k+1} = \psi(\boldsymbol{x}_k)$. If this series is convergent, it converges towards to this point. The centered inclusion function $\boldsymbol{\psi}(\boldsymbol{x})$ is given by:

$$\boldsymbol{\psi}(\boldsymbol{x}) = \psi(x_c) + \boldsymbol{J}_\psi(\boldsymbol{x} - x_c). \tag{2.36}$$

where $\boldsymbol{J}_\psi$ is the inclusion function of the Jacobian matrix of ψ and the x_c is the center of the $\boldsymbol{x}$ (equivalent to $mid(\boldsymbol{x})$). That allows to write down the Krawczyk contractor:

$$\mathcal{C}_k : \boldsymbol{x}_{k+1} = \boldsymbol{x}_k \cap (\psi(x_c) + \boldsymbol{J}_\psi(\boldsymbol{x} - x_c)). \tag{2.37}$$

Now we introduce $\psi(\boldsymbol{x}) = \boldsymbol{x} - Mf(\boldsymbol{x})$ which gives:

$$\mathcal{C}_K : \boldsymbol{x}_{k+1} = \boldsymbol{x}_k \cap (x_c - Mf(x_c) + (I - M\boldsymbol{J_f}(\boldsymbol{x}))(\boldsymbol{x} - x_c)). \tag{2.38}$$

Notice that I is the identity matrix. In practice, it is suggested to use the inverse $J_f^{-1}(x_c)$ which is a kind of preconditioning operations. The algorithm is as below:

Algorithm 4 Algorithm Krawczyk contractor $\mathcal{C}_K(\mathcal{X}, \mathcal{D}_{\boldsymbol{x}}, f(\cdot))$

Input: $\mathcal{D}_{\boldsymbol{x}}, f(\cdot)$;
Output: $[\boldsymbol{x}]$;
1: $M = J_f^{-1}(x_c)$;
2: $\boldsymbol{J}_\psi = I - M\boldsymbol{J}_f(\boldsymbol{x})$;
3: $\boldsymbol{i} = x_c - Mf(x_c) + (I - M\boldsymbol{J}_f(\boldsymbol{x}))(\boldsymbol{x} - x_c)$;
4: $\boldsymbol{x} = \boldsymbol{x} \cap \boldsymbol{i}$.

This contractor can be nested, which means that it can be used recursively $\mathcal{C}_K(\mathcal{C}_K(...\mathcal{C}_K(\boldsymbol{x})))$ until convergence towards the solution or until the stop point predefined.

Newton contractor ($\mathcal{C}_N$) This contractor is also for the *CSP* $\mathcal{H} : (\mathcal{X}, \mathcal{D}, f(\boldsymbol{x}) = 0)$. $f(\bullet)$ should be differentiable, the number of constraints N_c should be equal to the dimension of $\boldsymbol{x}$ which is also N_x, i.e. we have $N_x = N_c$.

In this contractor, first order Taylor expansion is used to get a fixed-point sub-solver $\psi(\boldsymbol{x}) = \boldsymbol{x} - J_f^{-1}(\boldsymbol{x}) * f(\boldsymbol{x})$. The inclusion function is then:

$$\boldsymbol{\psi}(\boldsymbol{x}) = \psi(x_c) + \boldsymbol{J}_f^{-1}(\boldsymbol{x})\boldsymbol{f}(\boldsymbol{x}). \tag{2.39}$$

Algorithm 5 Algorithm Newton contractor $\mathcal{C}_N(\mathcal{X}, \mathcal{D}_{\boldsymbol{x}}, f(\cdot))$

Input: $\mathcal{D}_{\boldsymbol{x}}, f(\cdot)$;
Output: $\boldsymbol{x}$;
1: $\boldsymbol{A} = \boldsymbol{J}_f(\boldsymbol{x})$;
2: $\boldsymbol{p} = \boldsymbol{x} - x_c$;
3: $\mathcal{C}_{GS}(\boldsymbol{A}\boldsymbol{p} + f(x_c) = 0)$;
4: $\boldsymbol{x} = \boldsymbol{x} \cap (\boldsymbol{p} + x_c)$.

The Newton contractor is given as:

$$\mathcal{C}_N : \boldsymbol{x}_{k+1} = \boldsymbol{x}_k \cap (\psi(x_c) + \boldsymbol{J}_f^{-1}(\boldsymbol{x})\boldsymbol{f}(\boldsymbol{x})). \tag{2.40}$$

This contractor can be nested, which means that it can be used recursively $\mathcal{C}_N(\mathcal{C}_N(...\mathcal{C}_N(\boldsymbol{x})))$ until convergence towards the solutions.

2.3.2.5 Examples and comparison

Different contractors adapt different situations, they all have advantages and limits. To show their performances, three examples are used and simulated numerically in Matlab.

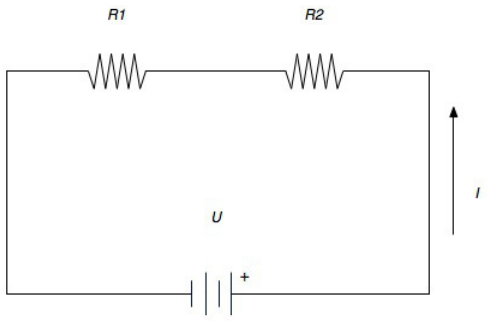

Figure 2.4: Electrical circuit example

Example 2.12 *Given an electrical circuit which contains two resistances, obtained following constraints:*

$$\mathcal{C} : \begin{cases} P = UI, \\ U = (R_1 + R_2) * I, \\ U_1 = R_1 * I, \\ U_2 = R_2 * I, \\ U = U_1 + U_2. \end{cases} \tag{2.41}$$

where $U \in [23, 26]V$,$I \in [4, 8]A$,$U_1 \in [10, 11]V$,$U_2 \in [14, 17]V$, *and* $P \in [124, 130]W$.

Table 2.1: Comparison result for example 2.12

	Nb-cycle	Results
$\mathcal{C}_P$	3	$\mathcal{X} = \begin{Bmatrix} [1.8462, 2.3065] \\ [2.5846, 3.3548] \end{Bmatrix}$
$\mathcal{C}_{FB}$	2	$\mathcal{X} = \begin{Bmatrix} [1.8462, 2.3065] \\ [2.5846, 3.3548] \end{Bmatrix}$

Contract $\mathcal{X} = \{\boldsymbol{R_1}, \boldsymbol{R_2}\}$, given the initial domains:

$$\mathcal{D} : \begin{cases} R_1 \in [0, \inf], \\ R_2 \in [0, \inf]. \end{cases}$$

For this example, the constraint propagation contractor and forward-backward contractor can be used; the results are given in table 2.1.

Nb-cycle represents the number of iterations of contraction algorithm until it achieves the stop point. The Table 2.1 shows that both contractors can achieve a good result; the forward-backward contractor has less iterations, as it repeats contraction equations twice (though in different direction) at each iteration.

Example 2.13 *Consider an equation as a constraint:*

$$\mathcal{C} : \{\boldsymbol{Ax} - \boldsymbol{b} = 0\}. \tag{2.42}$$

where

$$\boldsymbol{A} = \begin{Bmatrix} [4,5] & [-1,1] & [1.5,2.5] \\ [-0.5,0.5] & [-7,-5] & [1,2] \\ [-1.5,-0.5] & [-0.7,-0.5] & [2,3] \end{Bmatrix}, \boldsymbol{B} = ([3,4] \quad [0,2] \quad [3,4])^T \tag{2.43}$$

Contract $\mathcal{X} = \{\boldsymbol{x}\}$, with initial domain:

$$\mathcal{D}_1 : \boldsymbol{x} \in ([-10,10] \quad [-10,10] \quad [-10,10])^T$$

For this initial domain, all the contractors can be used, the results are presented in table 2.2:

The results show that the best result is obtained by the Gauss-Seidel algorithm with preconditioning.

Another initial domain is not bounded,

$$\mathcal{D}_2 : \boldsymbol{x} \in ([-\inf, \inf] \quad [-\inf, \inf] \quad [-\inf, \inf])^T.$$

Only gauss elimination can be used, the results are in table 2.3:

Gauss elimination with preconditioning provides better results compared to simple gauss elimination.

Table 2.2: Comparison result for example 2.13

	Nb-cycle	Results
$\mathcal{C}_P$	81	$\mathcal{X} = \begin{Bmatrix} [-2.0846, 1.6368] \\ [-0.7370, 1.7436] \\ [-0.3214, 3.8378] \end{Bmatrix}$
$\mathcal{C}_{FB}$	81	$\mathcal{X} = \begin{Bmatrix} [-2.0846, 1.6368] \\ [-0.7370, 1.7436] \\ [-0.3214, 3.8378] \end{Bmatrix}$
$\mathcal{C}_{GE}$	1	$\mathcal{X} = \begin{Bmatrix} [-1.8193, 1.1687] \\ [-0.4141, 1.7252] \\ [0.7002, 3.4208] \end{Bmatrix}$
$\mathcal{C}_{GE}$ with preconditioning	1	$\mathcal{X} = \begin{Bmatrix} [-1.1070, 1.3837] \\ [-0.7852, 1.3320] \\ [0.7584, 2.9841] \end{Bmatrix}$
$\mathcal{C}_{GS}$	45	$\mathcal{X} = \begin{Bmatrix} [-2.0845, 1.6367] \\ [-0.7370, 1.7436] \\ [-0.3214, 3.8378] \end{Bmatrix}$
$\mathcal{C}_{GS}$ with preconditioning	11	$\mathcal{X} = \begin{Bmatrix} [-1.1070, 1.3837] \\ [-0.8114, 1.3320] \\ [0.5757, 2.9841] \end{Bmatrix}$
$\mathcal{C}_K$	13	$\mathcal{X} = \begin{Bmatrix} [-10.0000, 10.0000] \\ [-9.6789, 9.2992] \\ [-10.0000, 10.0000] \end{Bmatrix}$

Table 2.3: Comparison result for example 2.13

	Nb-cycle	Results
$\mathcal{C}_{GE}$	1	$\mathcal{X} = \begin{Bmatrix} [-1.8193, 1.1687] \\ [-0.4141, 1.7252] \\ [0.7002, 3.4208] \end{Bmatrix}$
$\mathcal{C}_{GE}$ with preconditioning	1	$\mathcal{X} = \begin{Bmatrix} [-1.1070, 1.3837] \\ [-0.7852, 1.3320] \\ [0.7584, 2.9841] \end{Bmatrix}$

Table 2.4: Comparison result for example 2.14

	Nb-cycle	Results
$\mathcal{C}_P$	4	$\mathcal{X} = \begin{Bmatrix} [0.7584, 2.9841] * 10e^{-12} \\ [-0.0964, 0.2403] * 10e^{-12} \end{Bmatrix}$
$\mathcal{C}_{FB}$	3	$\mathcal{X} = \begin{Bmatrix} [0, 0.0000] * 10e^{-12} \\ [0, 0.2328] * 10e^{-12} \end{Bmatrix}$
$\mathcal{C}_K$	4	$\mathcal{X} = \begin{Bmatrix} [-0.7762, 0.7762] * 10e^{-10} \\ [-0.3342, 0.3342] * 10e^{-10} \end{Bmatrix}$

Example 2.14 *Given an equation in the form of following constraints:*

$$\mathcal{C} : \begin{cases} f_1(\boldsymbol{x}_1, \boldsymbol{x}_2) = \boldsymbol{x}_1^2 - 4\boldsymbol{x}_2. \\ f_2(\boldsymbol{x}_1, \boldsymbol{x}_2) = \boldsymbol{x}_2^2 - 2\boldsymbol{x}_1 + 4\boldsymbol{x}_2. \end{cases} \tag{2.44}$$

contract $\mathcal{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2\}$*, with the initial domain:*

$$\mathcal{D} : \begin{cases} \boldsymbol{x}_1 \in [-0.1, 0.1], \\ \boldsymbol{x}_2 \in [-0.1, 0.3]. \end{cases}$$

The results are given in table 2.4.

We can notice that in this example, the constraint propagation contractor gives the best result.

In conclusion, interval-based constraint propagation is an efficient way to reduce the initial domain towards the solution set. It is essential to speed up the interval analysis algorithm. The basic constraint propagation contractor and its forward-backward enforcement iteration provide generally good and quick results in linear discrete system. These are the prior choice for us.

2.4 Interval matrices

2.4.1 Inversion problem

The resolution of interval linear systems like $\boldsymbol{A}X = \boldsymbol{b}$ can be tricky because of the interval matrix $\boldsymbol{A}$. A numerical matrix may have singularities so that the inverse of the set diverges very quickly.

A square matrix A of n order is called *invertible* or *regular* when it exists a matrix A^* which has the same order as A and satisfies:

$$AA^* = A^*A = I_n,$$

where I_n is the identity matrix of order n, and the multiplication here is the ordinary multiplication of matrices. A square matrix is singular if and only if its determinant is 0, and in such case, the matrix is not invertible. An interval matrix is more likely to have singularities as the candidate matrices within bound are infinite. Traditionally, we assume the interval matrix to be square and regular, meaning that every matrix in the set is nonsingular. The exact result of the set $\boldsymbol{A}^{-1}$ is not necessarily convex, so it is more natural to define the inverse of an interval matrix as the narrowest interval matrix which contains the set $\boldsymbol{A}^{-1}$. As today, the algorithm used to approximate interval matrix inversion is proposed by Rohn in [Rohn 1993], and it is used in many numerical solvers and toolboxes like INTLAB [Rump 1999]. This algorithm assumes that the interval matrix is regular.

2.4.2 Regularity test

The regularity of interval matrices is very important for solving systems of linear interval equations. An interval matrix is required to be regular before the inversion operation.

An interval matrix is said to be regular if all scalar matrix candidates that belong to the set have non-null determinant and it is called singular otherwise. There exist lots of criteria for testing the regularity of interval matrices. Rump presents the criterion to verify positive definiteness of a symmetric or Hermitian matrix in [Rump 2006]. In [Rohn 2009] Rohn presents forty criteria for regularity test for interval matrices, some of which are efficient in practice. [Paşca 2010] presents the formerly verified condition for the regularity of an interval matrix.

Definition 2.17 *An interval matrix $\boldsymbol{A} \in \mathbb{IR}^{n\times n}$ is called regular if and only if it satisfies:*

$$\forall A \in \boldsymbol{A}, \det A \neq 0,$$

otherwise it is called singular:

$$\exists A \in \boldsymbol{A}, \det A = 0.$$

2.4.3 Rohn's approach

Rohn uses the centered form of interval matrix [1] and a set of ± 1 vectors Y to generate the possible vertices of $\boldsymbol{A}$ by

$$A_{yz} = A_c - T_y \Delta T_z,$$

the complexity order is at most 2^{2n-1}. It still does not exhaust all the vertices of $\boldsymbol{A}$ the number of which equals 2^{n^2}. Then some (but not necessarily optimal) bounds of the inverse interval matrix are defined as:

$$\underline{\boldsymbol{B}}_{i,j} = min(A_{yz}^{-1})_{i,j}, y, z \in Y,$$

1. $\boldsymbol{A} = A_c + \Delta, A_c = mid(\boldsymbol{A}), \Delta = rad(\boldsymbol{A})$

and

$$\overline{\boldsymbol{B}}_{i,j} = max(A_{yz}^{-1})_{i,j}, y, z \in Y,$$

then $\boldsymbol{A}^{-1}$ is approximated by $[\underline{\boldsymbol{B}}, \overline{\boldsymbol{B}}]$. The complete deduction of this approximation can be found in [Rohn 1993].

This approach is widely used. It can provide a good approximation of an interval matrix inverse as long as the matrix is regular. If any of the vertices of the original interval matrix has null determinant, the result is not usable. Another issue is that the approximation is rough and in certain circumstance it may exacerbate the pessimism due to interval arithmetic.

2.4.4 Inverse interval matrix with SIVIA

It has been shown in the previous section that the algorithm SIVIA can be solved to find the set inversion problem for interval functions. The precision of the result can be adjusted according to the requirements. We can either have a narrower interval container or less computational time.

In our work, the interval matrix that needs to be inverted is square and symmetric ($H_k \hat{P}_{k+1|k} H_k^T + R_k$). According the matrix property, the inverse symmetric matrix is still symmetric. Finding the matrix inverse can be interpreted as finding the solution for a family of constraints. Moreover, finding a solution for a linear system like $\boldsymbol{AX} = \boldsymbol{B}$, whether the matrix $\boldsymbol{A}$ is square or not, can always be treated as constraints family.

Given an interval matrix $\boldsymbol{A} \in \mathbb{IR}^{n \times n}$ and $\boldsymbol{B} \in \mathbb{IR}^{n \times m}$, if the solution of $\boldsymbol{X}$ exists, it must satisfy the following constraints:

$$\mathcal{C}_{i,j} : \sum_{p=1}^{n} a_{i,p} x_{p,j} = b_{i,j}, i = \{1, \cdots, n\}, j = \{1, \cdots, m\}. \tag{2.45}$$

Each constraint can be solved by SIVIA algorithm, the complexity of this approach is defined by following equation, based on the Equation (2.25):

$$i \times \left(\frac{w(\boldsymbol{x}(0))}{\varepsilon} + 1 \right)^j, \tag{2.46}$$

where the $\boldsymbol{x}$ is the vector box constructed from ith row of matrix $\boldsymbol{X}$:

$$\boldsymbol{X}_{i,1} \times \boldsymbol{X}_{i,2} \times \cdots \times \boldsymbol{X}_{i,j}.$$

2.4.4.1 Application example

Example 2.15 *Given*

$$\boldsymbol{AX} = \boldsymbol{B}, \boldsymbol{A} = \begin{pmatrix} [3.7, 4.3] & [-1.5, -0.5] \\ [-1.5, -0.5] & [3.7, 4.3] \end{pmatrix}, \boldsymbol{B} = \begin{pmatrix} [-14, 0] \\ [-9, 0] \end{pmatrix}.$$

find solution set of $\boldsymbol{X}$.

Rohn's approach : $\boldsymbol{X} = \boldsymbol{A}^{-1} \times \boldsymbol{B} = \begin{pmatrix} [-5.7163, 1.3829] \\ [-4.7546, 1.4212] \end{pmatrix}$

Inversion by SIVIA : The initial search space can be set as the result of Rohn's approach (see Figure 2.5); the reason is that Rohn's approach is already a sub-optimal solution. It is over-estimated and the stop factor of SIVIA ε is set to 0.01. The result is shown in Figure 2.5.

Notice that in the figure, the square of entire space is the result of Rohn's approach. In the inversion using SIVIA, the green area is the part such that $\forall A \in \boldsymbol{A}, A\boldsymbol{X} \subset \boldsymbol{B}$, while in yellow area, $\exists A \in \boldsymbol{A}, A\boldsymbol{X} \subset \boldsymbol{B}$, and the red area corresponds to the rejected solutions. It is very clear that the result obtained by our approach is more accurate.

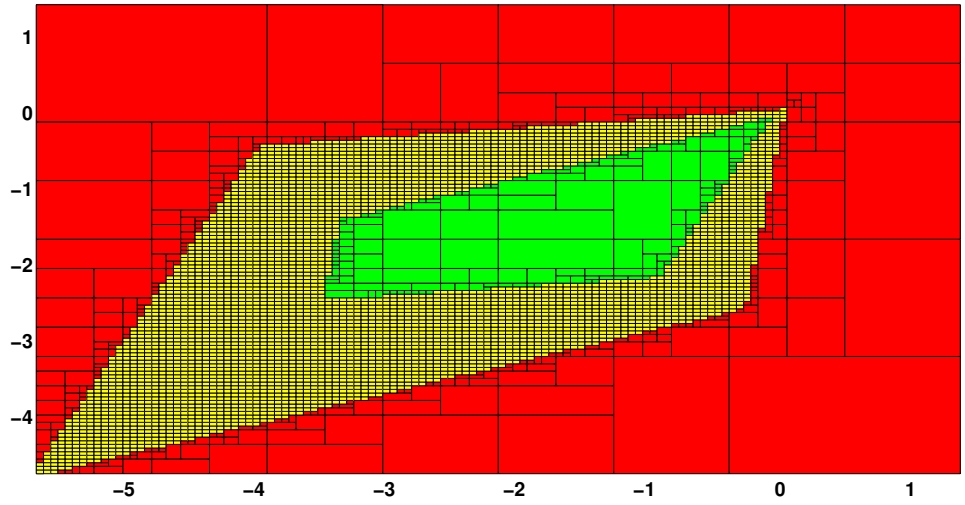

Figure 2.5: Matrix inversion by SIVIA

2.5 INTLAB

There exist different libraries of interval arithmetic for numerical implementation and simulation. Some of them even provide complete solvers for certain algorithms, such as contractor solvers from Ibex[Jaulin 2008]. Our simulation environment is under MATLAB [Hargreaves 2002], the toolbox INTLAB meets our criterion of usage [Rump 1999]. The philosophy of INTLAB is that everything is written in Matlab code to assure best portability. Throughout our use, we find that INTLAB is very flexible as we can write our own solvers under MATLAB easily, and run test in a simple and efficient way. The drawback of this choice is that almost every solver should be written by our own, and the calculation speed is limited and needs lots of delicate optimizations.

The whole work is written in Matlab scripts with the INTLAB toolbox.

2.6 Conclusion

In this chapter, the basic notions of interval arithmetic are introduced. One important point of interval arithmetic is to find a minimum inclusion function, which can reduce the pessimism.

The techniques that can tighten the interval calculation result are also discussed in this part. It concerns two major methods. The first one is the set inversion approach which can provide a better enclosure of result, even in the non-convex case; the algorithm needs more time and resource to obtain the result. In certain applications, another technique , the constraint propagation, is more practical. This approach looks for the constraints in equations and variables' relations, different methods are used to proceed the search space contraction.

These two approaches can be used in the same application. The constraint propagation provides a smaller initial search space for the set inversion technique, the latter gives a refined result, as it is time consuming, a smaller initial search space can reduce the calculation time and resource.

The interval matrix is highlighted in this chapter. Especially the inversion problem of interval matrix is discussed. The objective is to find singularities in an interval matrix, and find the matrix inverse without losing any solution, but still has a good precision and speed. The set inversion algorithm can be used to obtain the inverse matrix without inversion operation.

Lastly, the toolbox INTLAB is introduced as it is used widely in this work.

Chapter III
Uncertain stochastic systems

CHAPTER 3

Uncertain stochastic systems

Contents

3.1 Introduction

The set-membership approach can provide guaranteed solution for dealing with bounded uncertainties. Numerous related works have been developed in the last few years. The characteristic of this approach are revealed in comparison with conventional stochastic approach.

The primary differences between the set-membership approach and the stochastic framework are presented in Table 3.1, inspired by the contribution established in [Gelso 2008].

Table 3.1: Differences between the stochastic approach and the set-membership approach

	Stochastic approach	Set-membership approach
Uncertainties	Process and measurement noise respect a probabilistic model	Bounded value representative otherwise unknown
Model	Sensitive to strong non-linearity	Not sensitive to strong non-linearity
Challenge	Modelling complexity	Computational complexity, over estimation

Parameters and state estimations from experimental measures are usually obtained within a stochastic framework in which assumed known distribution laws are associated to interferences and noisy measurements. In contrast, in the bounded error context, measurements and modelling uncertainties are supposed to be unknown but stay within known bounds. It is not always possible to get enough information on disturbances and noises. Lack of information may turn the usual stochastic framework inappropriate. In such cases, the model of bounded uncertainties can be a better solution.

Regarding the computational complexity, if all the pre-required statistical informations are given, the stochastic framework is generally less difficult than the set-membership approach. But in cases where the distribution law is unknown, set-membership methods are more adequate. In this term, the stochastic method requires more prior information than the set-membership method. Both approaches can obtain similar estimation results for a proper combination of tuning parameters, as mentioned in [Gelso 2008].

In addition to the difficulty of setting initial conditions, the set-membership approach has overestimation drawbacks, and may fail estimation and detection process. A stochastic method however, would not invalidate the non relevant fault models but it would conclude to the existence of a solution with a confidence interval, which is difficult to interpret at some time.

As both techniques have specific advantages and disadvantages, they may interact synergically. A stochastic method can provide means for analyzing the properties of a set estimator, whereas a set-membership technique can provide the initial entry to a system

without assuming the distribution law of uncertainty in advance. They are hence more complementary than competitive.

3.2 Definition and formulation

Based on above idea, it is interesting to combine these two uncertainty models in one system, which we call *Uncertain Stochastic System*. The name emphasizes that measurement and system noises are still modelled under a statistical framework, meanwhile bounded uncertainties exist in system parameters which are given set values.

As mentioned previously, in this work we mainly concentrate on linear dynamic discrete system. It is called Kalman model [Kalman 1960] given by the following form:

$$\begin{cases} x_{k+1} = Ax_k + Bu_k + w_k, \\ y_k = Cx_k + Du_k + v_k, k = 0, 1, 2, ... \end{cases} \tag{3.1}$$

where $x_k \in \mathbb{R}^n$, $y_k \in \mathbb{R}^m$, $u_k \in \mathbb{R}^p$ are state observation and input vectors, respectively, $A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times p}, C \in \mathbb{R}^{m\times n}$ and $D \in \mathbb{R}^{m\times p}$ are constants matrices, $\{w_k\}$ and $\{v_k\}$ are independent Gaussian white noises sequences, with covariance matrices Q and R, respectively, definite positive by definition:

$$\begin{array}{l} E\{w_k, w_l\} = Q\delta_{kl}, E\{v_k, v_l\} = R\delta_{kl}, \\ E\{w_k, v_l\} = E\{w_k, x_0\} = E\{v_k, x_0\} = 0, \\ \forall k, l = 0, 1, 2, ... \end{array}$$

where δ_{kl} is the Kronecker symbol.

For this system, Kalman filtering and recursive least square method are well known to provide optimal estimation on unknown states by taking account measurements at each iteration.

In order to adapt the Uncertain Stochastic System model, system (3.1) is reformulated in this double context,

$$\begin{cases} x_{k+1} = Ax_k + Bu_k + w_k, A \in \boldsymbol{A}, B \in \boldsymbol{B}, \\ y_k = Cx_k + Du_k + v_k, \quad C \in \boldsymbol{C}, D \in \boldsymbol{D}, \\ \boldsymbol{A} \in \mathbb{IR}^{n\times n}, \boldsymbol{B} \in \mathbb{IR}^{n\times p}, \boldsymbol{C} \in \mathbb{IR}^{m\times n}, \boldsymbol{D} \in \mathbb{IR}^{m\times p}, \ \ k = 0, 1, 2, ... \end{cases} \tag{3.2}$$

In calculus, involved parameters are belonging to the boxes indicated in **bold** type. Notice that in the system model, states and observations are not set values. At time k the parameters' values are certainly punctual but unfortunately unknown, however it is assumed that they belong to bounded areas thus we can use the set value approximation to replace the unknown exact values in the following sections. It is also assumed that parameters are time invariant.

3.3 Estimation based on the initial state and historical command and noises

The mixture of different types of variables makes the system analysis difficult. Neither the statistical approach nor the set-membership method has considered the presence of various uncertainty approximations in one framework. Only a part of the state is biased by known statistical noises.

Based on system (3.2) and the alternative form of interval definition in Equation (2.1), the following equations are obtained:

$$\begin{cases} x_{k+1} = (A + \Delta A)x_k + (B + \Delta B)u_k + w_k, \\ y_k = (C + \Delta C)x_k + (D + \Delta D)u_k + v_k, k = 0, 1, 2, ... \end{cases} \tag{3.3}$$

Interval parameters are presented as a combination of a nominal value described in equation (3.1) and a bounded uncertainty term.

At time k, the state estimate $\hat{x}_k$ can be evaluated from x_0 by:

$$\hat{x}_k = (A + \Delta A)^k x_0 + \sum_{i=1}^{k} (A + \Delta A)^{k-i} w_{i-1} + \sum_{i=1}^{k} (A + \Delta A)^{k-i} (B + \Delta B)u_{i-1},\ k \geqslant 0. \tag{3.4}$$

this is demonstrated in Appendix A.

Then $\hat{x}_k$ can be divided into $\hat{x}^S$ and $\hat{x}_k^I$,

$$\begin{cases} \hat{x}_k = \hat{x}_k^S + \hat{x}_k^I, \\ \hat{x}_k^S = \sum_{i=1}^{k} (A + \Delta A)^{k-i} w_{i-1}, \\ \hat{x}_k^I = (A + \Delta A)^k x_0 + \sum_{i=1}^{k} (A + \Delta A)^{k-i} (B + \Delta B)u_{i-1}. \end{cases} \tag{3.5}$$

$\hat{x}_k^S$ is the sub-state biased by the statistical noise and $\hat{x}_k^I$ contains only bounded uncertainties.

It is also possible to divide into two parts:

$$\begin{aligned} \hat{y}_k &= (C + \Delta C)\hat{x}_k + (D + \Delta D)u_k + v_k \\ &= (C + \Delta C)(\hat{x}_k^S + \hat{x}_k^I) + (D + \Delta D)u_k + v_k \\ &= \left((C + \Delta C)\hat{x}_k^S + v_k\right) + \left((C + \Delta C)\hat{x}_k^I + (D + \Delta D)u_k\right) \\ &= \hat{y}_k^S + \hat{y}_k^I. \end{aligned} \tag{3.6}$$

therefore:

$$\begin{cases} \hat{y}_k = \hat{y}_k^S + \hat{y}_k^I, \\ \hat{y}_k^S = (C + \Delta C) \sum_{i=1}^{k} (A + \Delta A)^{k-i} w_{i-1} + v_k, \\ \hat{y}_k^I = (C + \Delta C) \left[(A + \Delta A)^k x_0 + \sum_{i=1}^{k} (A + \Delta A)^{k-i} (B + \Delta B) u_{i-1} \right] + (D + \Delta D) u_k. \end{cases} \tag{3.7}$$

The same comments as far $\hat{x}_k^s$ and $\hat{x}_k^I$ hold for $\hat{y}_k^s$ and $\hat{y}_k^I$.

3.4 Uncertainty in probability density functions

The previous section shows that, in any way, the statistical noise and the uncertain parameters can not be separated completely; therefore they can not be treated as parallel processes. It is then interesting to investigate the influence of the set valued parameters in a conventional stochastic model.

Case 1 Firstly we consider a discrete state model without noise and uncertainty,

$$x_k = A x_{k-1}. \tag{3.8}$$

The state estimate at time k based on previous estimate at time $k-1$ can be easily yielded by

$$\hat{x}_k = A \hat{x}_{k-1}, \tag{3.9}$$

as it is illustrated in Figure 3.1 (up-left). Without noise and other uncertainty, the probability of the state passing from $\hat{x}_{k-1}$ to $\hat{x}_k$ equals to 1.

Case 2 Next, a statistical noise w is added. The original system turns into:

$$x_k = A x_{k-1} + w_{k-1}. \tag{3.10}$$

We consider a centered noise which has a normal distribution, $w \sim N(0, \sigma^2)$.

This classical system model can be solved by numerous estimation methods. One can use correction gain to minimize the error co-variance of the estimate by using the system output measurement. A correction gain G is designed to satisfy some criterion in order to obtain an optimal estimate.

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + G(y - \hat{y}_{k|k-1}).$$

where $\hat{x}_{k|k}$ is the *a posteriori* estimate, $\hat{x}_{k|k-1}$ is the *a priori* estimate, $\hat{y}_{k|k-1}$ is the *a priori* output prediction.

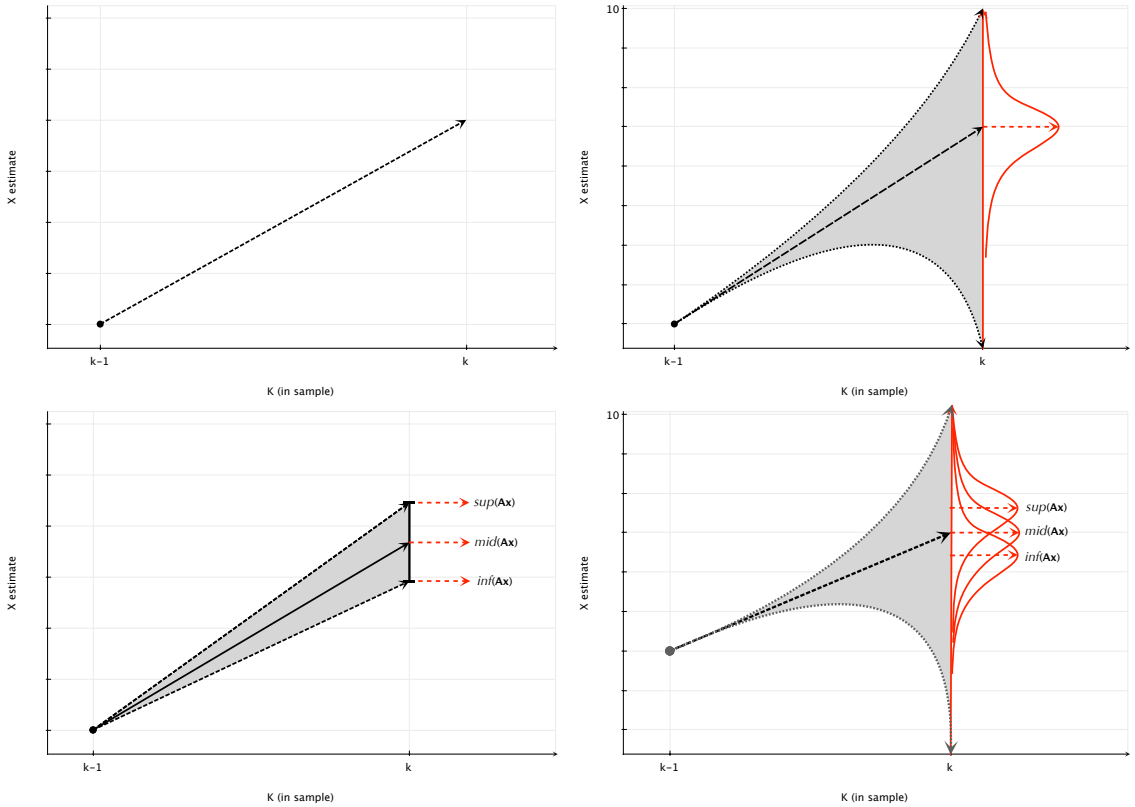

Figure 3.1: Case 1: state estimation without noise and uncertainty (up-left); Case 2: state estimation with noise but without uncertainty (up-right); Case 3: state estimation without noise but with uncertainty (bottom-left); Case 4: state estimation with noise and uncertainty (bottom-right)

In this case, the estimate is an optimal approximation which is the mode. Since we can not know the exact value of the noise at each time, the only information that we have is the distribution of all possible system state estimates and the probability of each possible noise value defined by the probability density function (3.12) on an infinite support. The set of possible state estimates can be illustrated by the Figure 3.1 (up-right).

Notice that in this case, the possible state is not bounded as the possible noise values are normally distributed in $\mathbb{R}$. For the mean value μ of the noise, the state has the theoretical largest probability among the set bunch, though the state estimate is yielded by other criterion such as minimum error covariance, they are not necessarily the same.

Case 3 If the system does not involve statistical noise but has parameter uncertainty, the system estimate turns into a bounded set of values:

$$x_k = \boldsymbol{A} x_{k-1}. \tag{3.11}$$

All the values in the set have equal probability. They are uniformly distributed on a

bounded support. It is illustrated by the Figure 3.1 (bottom-left).

Case 4 Finally we consider parameter uncertainty and statistical noise at the same time. In this situation, the result is a set of estimates which can be seen as two parts. The first part is yielded from the parameters uncertainty and corresponds to a set of nominal estimates without noises perturbation. The second part is the statistical noise which is added to each estimate obtained in the first part. The result is in fact a family of statistical estimates as it is illustrated in the Figure 3.1 (bottom-right). The mean value and standard derivation of this set can be expressed as a bounded interval:

$$\mu \in [\underline{\boldsymbol{\mu}}, \overline{\boldsymbol{\mu}}] = \boldsymbol{\mu},$$

which will be explained more explicitly in the next section.

It raises the following question: **How can we interpret a statistical noise with interval mean value?** How does bounded uncertainty affect the basic characteristics of a stochastic framework, for instance, the probability density function? If the original noise is Gaussian with interval mean value, is the family of a set of Gaussian noises still Gaussian? In the set-membership context, the expectation and the variance of a random data series needs to be carefully investigated.

3.4.1 Interval expectation and variance

As seen in the last section, if bounded uncertainties are added to a statistical gaussian error, the characteristic of the random data changes. Let us consider the case of a normal distribution parametrized in terms of the mean μ and the variance σ, the family of densities is described as:

$$f(x;\mu,\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} exp\left(-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right). \tag{3.12}$$

When the variable x is represented by a set value, $x \in \boldsymbol{x}$, every candidate value in the set is also assumed to have statistical distribution, which yields a set of statistical variables, whose average values are included in the set $\boldsymbol{\mu}$ and the standard Deviation in the set $\boldsymbol{\sigma}$. Is the family of statistical variables still following the same statistical distribution? In order to investigate the characteristic of such variable family, both curve fitting simulation and theoretical demonstration are proposed in the following sections.

3.4.1.1 Set valued μ et σ simulation

Considering an example $\boldsymbol{x}_k \sim N(\boldsymbol{\mu}, \boldsymbol{\sigma}^2)$, μ and σ are assumed to belong to $\boldsymbol{\mu}, \boldsymbol{\sigma}$. Initially we can take $\mu = 1, \sigma = 2$ and $x_k \sim N(1,4)$.

Situation 1: $\boldsymbol{\mu} = [0,2]$, and σ stays punctual, the $\boldsymbol{x}_k \sim N([0,2],4)$ We simulated $\boldsymbol{\mu} = [0,2]$ under Matlab: 1000 samples of μ uniformly distributed in the interval $\boldsymbol{\mu}$, and for each μ we consider 100 samples of $\boldsymbol{x}_k$, which gives us a total $100 * 1000$ random values, they follow the normal distribution with μ defined as an interval value.

The set of all the value is then fitted by a new distribution function, which has a mean value at 1.0033, variance at 2.0807. We obtain $\boldsymbol{x}_k \sim N([0,2],4) \simeq x_k \sim N(1.0033, 4.3294)$, which means the set of all values with an interval mean value, still follows a normal distribution law.

The figure is given in Figure 3.2.

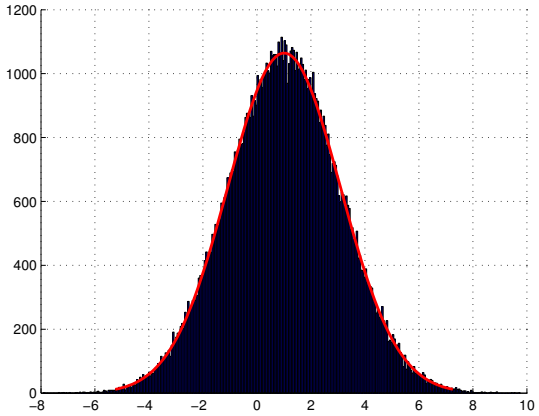

Figure 3.2: $\mu \in [0,2]$, σ fixed

Situation 2: $\boldsymbol{\sigma} = [1,3]$, μ stays punctual, $\boldsymbol{x}_k \sim N(1,[1,9])$ We simulated $\boldsymbol{\sigma} = [1,3]$ under Matlab: 1000 samples of σ uniformly distributed in the interval $\boldsymbol{\sigma}$, and for each σ we give 100 samples of $\boldsymbol{x}_k$, in total $100 * 1000$ random values; they follow the normal distribution with σ defined as an interval value.

The set of all the value is then fitted by a new distribution function, which has a mean value at 0.9953, variance at 2.0809, approximately we obtain $\boldsymbol{x}_k \sim N(1,[1,9]) \simeq x_k \sim N(1.0033, 4.3294)$, which means the set of all values with an interval mean value, globally still follows the normal distribution law.

The figure is given in Figure 3.3.

Situation 3: $\boldsymbol{\sigma} = [1,3]$, $\boldsymbol{\mu} = [0,2]$, $\boldsymbol{x}_k \sim N([0,2],[1,9])$ We simulated $\boldsymbol{\mu} = [0,2]$ and $\boldsymbol{\sigma} = [1,3]$ under Matlab, 1000 samples of μ uniformly distributed in the interval $\boldsymbol{\mu}$, 1000 samples of σ uniformly distributed in the interval $\boldsymbol{\sigma}$ and for each pair of μ and σ we give 100 samples of $\boldsymbol{x}_k$, which gives us a total $100 * 1000 * 1000$ random values, they follow the normal distribution with μ and σ defined as interval values.

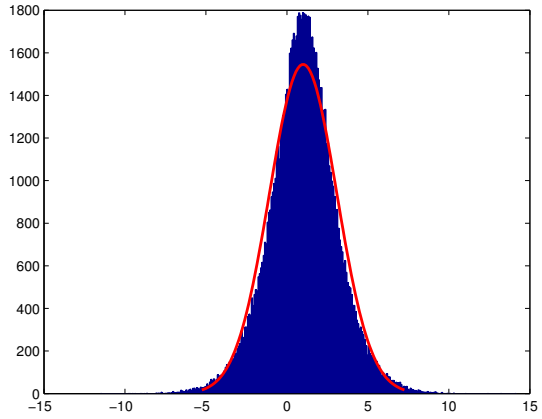

Figure 3.3: μ fixed, $\sigma \in [1, 3]$

The figure is given in 3.4.

The set of all the value is then fitted by a new distribution function, which has a mean value at 1.0147, variance at 2.0838, approximately we can obtain $\boldsymbol{x}_k \sim N([0, 2], [1, 9]) \simeq x_k \sim N(1.0147, 4.7602)$, which means the set of all values with an interval mean value, globally still follows the normal distribution law.

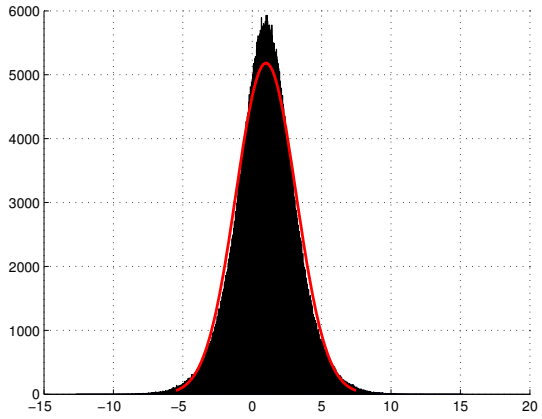

Figure 3.4: $\mu \in [0, 2], \sigma \in [1, 3]$

These tests conclude that, with interval mean value or/and variance, the set of random data still obeys a normal distribution law; the approximation curving shows that they tend to have the center of the interval as their statistical characteristics.

3.4.1.2 Set valued μ et σ demonstration

Given a random continuous variable x, following a normal distribution with mean value μ and variance σ^2, its density is represented by:

$$\boldsymbol{f}(x;\mu,\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} exp\left(-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right), \forall x \in \mathbb{R}.$$

We now consider the functions with mean μ and standard deviation σ, $\mu \in \boldsymbol{\mu}, \sigma \in \boldsymbol{\sigma}$, which implies the uncertain stochastic characteristic:

$$\boldsymbol{f}(x;\mu,\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} exp\left(-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right), \forall x \in \mathbb{R}, \mu \in \boldsymbol{\mu}, \sigma \in \boldsymbol{\sigma}. \tag{3.13}$$

We call this set of functions $\boldsymbol{f}(x, \boldsymbol{\mu}, \boldsymbol{\sigma}^2)$.

3.4.1.3 Interval linear function

If X is a variable of dimension N which follows the gaussian distribution, which means $X \sim N(m_x, V_x)$, does the variable Y defined by affine transformation,

$$Y = AX + b, A \in \boldsymbol{A}, b \in \boldsymbol{b}$$

follow a gaussian distribution.

Proposition 3.1 *Let X be a random continuous variable of dimension N with normal distribution characterized by $X \sim N(m_x, V_x)$,the variable Y evaluated from the affine transformation:*

$$Y = AX + b = g(X), A \in \boldsymbol{A}, b \in \boldsymbol{b}, \tag{3.14}$$

is also a random continuous variable with normal distribution characterized by

$$Y \sim N(Am_x + b, AV_xA^T).$$

if and only if A is invertible.

Proof 3.1 *From the Equation (3.14), we have:*

$$x = A^{-1}(y - b) = g^{-1}(y). \tag{3.15}$$

The Jacobian matrix of function $g^{-1}(y)$ can be represented:

$$J_{g^{-1}}(y) = A^{-1}.$$

The Jacobian determinant is:

$$|det(J_{g^{-1}}(y))| = |det(A^{-1})| = \frac{1}{det(A)}.$$

Given $f_X(x)$ the density function of variable x, and $f_Y(y)$ the density function of variable y, we then have:

$$f_Y(y) = \frac{1}{(det(A))} f_X(A^{-1}(y-b)). \tag{3.16}$$

We also have:

$$f_X(x) = \frac{1}{\sqrt{2\pi * det(V_x)}} e^{-\frac{(x-m_x)^T V_x^{-1}(x-m_x)}{2}}. \tag{3.17}$$

It yields:

$$m_y = E(y) = Am_x + b, \tag{3.18}$$

and

$$V_y = E((y-m_y)(y-m_y)^T), \tag{3.19}$$

where

$$y - m_y = Ax + b - Am_x - b = A(x-m_x),$$

so that:

$$\begin{aligned} V_y &= E((y-m_y)(y-m_y)^T) \\ &= E(A(x-m_x)(x-m_x)^T A^T) = AV_x A^T. \end{aligned} \tag{3.20}$$

3.4.1.4 Interval covariance matrix

For random vectors x_1 and x_2 (of dimension n) the $n \times n$ covariance matrix is equal to:

$$Cov(x_1, x_2) = E\left((x_1 - E(x_1))(x_2 - E(x_2))^T\right).$$

It can be extended to the set-membership context.

For random interval vectors $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ (of dimension n) the $n \times n$ interval covariance matrix is equal to:

$$Cov(\boldsymbol{x}_1, \boldsymbol{x}_2) = E\left((\boldsymbol{x}_1 - E(\boldsymbol{x}_1)(\boldsymbol{x}_2 - E(\boldsymbol{x}_2))^T\right). \tag{3.21}$$

In our work, the state vector is of dimension n at instance k, $\boldsymbol{x}_k = (\boldsymbol{x}_k^1, \boldsymbol{x}_k^2, ..., \boldsymbol{x}_k^n)^T$, $\boldsymbol{x}_k^i$ is the i_{th} component of interval vector $\boldsymbol{x}$ at time k; its covariance matrix is defined as $Cov(\boldsymbol{x}_k, \boldsymbol{x}_k)$.

3.5 Linear uncertain stochastic system state estimation

In this section, we present the set-membership state estimation problem for linear discrete-time interval systems based on the knowledge and notions obtained at previous sections and introduce the existing approaches. We remind that the system formulation is as following:

In order to adapt the Uncertain Stochastic System model, system (3.1) is reformulated with in this double context,

$$\begin{cases} x_{k+1} = Ax_k + Bu_k + w_k, A \in \boldsymbol{A}, B \in \boldsymbol{B}, \\ y_k = Cx_k + Du_k + v_k, C \in \boldsymbol{C}, D \in \boldsymbol{D}, \\ \boldsymbol{A} \in \mathbb{IR}^{n\times n}, \boldsymbol{B} \in \mathbb{IR}^{n\times p}, \boldsymbol{C} \in \mathbb{IR}^{m\times n}, \boldsymbol{D} \in \mathbb{IR}^{m\times p}, k = 0, 1, 2, ... \end{cases}$$

The problem is to estimate state of such system by considering the statistical noise and bounded parameter tolerance at the same time.

3.5.1 Interval recursive least squares method (IRLS)

In case the noises are not supposed to be independent ($E[w_i v_j^T] \neq 0$, $i = 0, 1, ...$, $j = 0, 1 \ldots$) or when the estimation of the covariance matrix is not credible, the methods based on statistical approximation are less efficient; to solve the problem, some method can still be used by not including directly the characteristic of statistical noise. First we explore the estimation approach based on recursive least square method. In this context, a filter called "interval recursive least squares (IRLS)" has been proposed in order to obtain a state estimation in the form of set value which enclosures the real states with high confidence [Li 2008].

By introducing a forgetting factor like in the conventional least squares filter, and by replacing the correspond matrix by intervals, the new algorithm is the following:

Initialization:

$$\hat{x}_0 = x_0, P_0 = \alpha I, \lambda. \tag{3.22}$$

Recursion:

$$\begin{aligned} &\hat{\boldsymbol{x}}_{k+1} = \boldsymbol{A}\hat{\boldsymbol{x}}_k + \boldsymbol{L}_{k+1}(\boldsymbol{y}_{k+1} - \boldsymbol{CA}\hat{\boldsymbol{x}}_k) \\ &\boldsymbol{P}_{k+1} = \lambda^{-1}(I_n - \boldsymbol{L}_{k+1}\boldsymbol{CA})\boldsymbol{P}_k\boldsymbol{A}^T, \\ &\boldsymbol{L}_{k+1} = \boldsymbol{P}_k\boldsymbol{A}^T\boldsymbol{C}^T(\lambda I_n + \boldsymbol{CAP}_k\boldsymbol{A}^T\boldsymbol{C}^T)^{-1} \\ &k = 1, 2, ... \end{aligned} \tag{3.23}$$

where $\alpha > 0$ is a parameter defined by the user, and λ is a forgetting factor such that $0 < \lambda \leqslant 1$.

A major issue in the set-membership context is the pessimism introduced by the interval arithmetic. Uncertainty is cumulated at each iteration and the interval matrix inversion is time consuming, sometime divergent. To reduce this pessimism, special techniques should be used. These techniques are developed in further subsection.

As this approach does not concern the statistical noise, it is used in this thesis as a comparison technique.

3.5.2 Interval Kalman filtering (IKF)

In the conventional sense, the Kalman Filter equations provide a minimum variance estimator over all unbiased estimators. In [Chen 1997], an algorithm based on interval conditional expectation for interval linear systems has been developed. It has the same structure as the conventional Kalman filter algorithm while preserving the statistical optimality and the recursive computational scheme in the set-membership context; some extensions for non-linear system are also developed using the same approach [Ashokaraj 2005, He 1999].

3.5.3 Sub-optimal Interval Kalman filter (sIKF)

In [Chen 1997], it is suggested to use the upper bound of interval matrices as a regular matrix to avoid singularity problems in interval matrix inversion. This point of view leads to a sub-optimal solution that does not preserve guaranteed results, some solutions being lost. We note **sIKF** for this algorithm (sub-optimal interval Kalman filtering).

3.6 The improved Interval Kalman filter (iIKF)

It can be noticed that the previous algorithms does not include recent advances in interval analysis and constraint propagation techniques. This is why we propose a new recursive estimator including these recent advances that deals better with matrix inversion and controls better over-estimation.

3.6.1 Set-value propagation in Kalman filtering

In the set-membership context, we remind that the matrices A, B, C and D are considered as interval matrices, noted $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$ and $\boldsymbol{D}$. Notice that $x_{0|0}, P_{0|0}, u_k, y_k$ could be boxes due to the measurement errors, instrument precision or incomplete system information. Since punctual value can be considered as interval values for which width is equal to zero, the system framework can be unified for both interval values and punctual values. In the following, we evaluate the impact of changes between the conventional sense and the interval approach of Kalman filtering. Let us note, $x_0 \triangleq \boldsymbol{x_0}, w(\boldsymbol{x}_0) = 0$. To simplify explanation, we consider that the matrix D in Equation (3.1) is equal to zero.

Initial value of states x_0 are considered statistically known with expectation m_0 and covariance P_0, these variables could be an interval.

The relation between x_0 and P_0 is defined as $x_0 \sim \mathrm{N}(m_0, P_0)$.

Noises are considered as statistically known, meaning covariance matrices Q and R defined in section 3.2 are punctual. But if information about statistic characteristic on noise is not sufficient, their covariance matrix could also be a box. In that case, more dedicated

work can be found as "imprecise probability" ([Walley 1991]), and it is also compatible in our framework.

It is important to know in the set-membership context if the classic estimation algorithm also holds. To verify this query we need to start from basic definitions to evaluate the impact of interval uncertainty.

Prediction Firstly the prediction step is investigated; the initial condition is given by:

$$\hat{x}_{0|0} = x_0, \hat{P}_{0|0} = P_0.$$

Calculus for the *a priori* state estimate vector is inherit directly from the deterministic model, while corresponding variables are replaced by boxes which indicate bounded uncertainty:

$$\hat{\boldsymbol{x}}_{k+1|k} = \boldsymbol{A}\hat{\boldsymbol{x}}_{k|k} + \boldsymbol{B}\boldsymbol{u}_k. \tag{3.24}$$

The Equation (3.24) gives all the possible *a priori* state estimates from set of parameters based on the previous state estimate and current system input. The estimate state is a box.

At the previous time, the estimation error is characterized by $\boldsymbol{P}_{k|k}$. The prediction model does not include noise so the estimation error at the present instant should also be updated:

$$\hat{\boldsymbol{P}}_{k+1|k} = \boldsymbol{A}\hat{\boldsymbol{P}}_{k|k}\boldsymbol{A}^T + Q. \tag{3.25}$$

This equation can be interpreted as all possible *a priori* estimation error covariances between real state and *a priori* state estimate at time $k+1$. It is important to verify if the classical error covariance holds under the set-membership context.

The definition of covariance matrix which indicates the estimated accuracy of the estimate has been rewritten according to the basic definition:

$$\boldsymbol{P}_{k|k} = E\left((x_k - \hat{\boldsymbol{x}}_{k|k})(x_k - \hat{\boldsymbol{x}}_{k|k})^T\right), \tag{3.26}$$

$$\boldsymbol{P}_{k+1|k} = E\left((x_{k+1} - \hat{\boldsymbol{x}}_{k+1|k})(x_{k+1} - \hat{\boldsymbol{x}}_{k+1|k})^T\right) \tag{3.27}$$

where $\boldsymbol{P}_{k|k}$ is the *a posteriori* error covariance and $\boldsymbol{P}_{k+1|k}$ is the prediction error covariance.

Another way to represent the matrix is element by element, which is written by following:

$$\boldsymbol{P}^{(i,j)} = E\left((x_i - \hat{\boldsymbol{x}}_i)(x_j - \hat{\boldsymbol{x}}_j)\right), \quad i = 1, ..., n, j = 1, ..., n.$$

where x and $\hat{\boldsymbol{x}}$ are vectors with n elements, denoted by $[x_1, x_2, ..., x_n]^T$ and $[\hat{\boldsymbol{x}}_1, \hat{\boldsymbol{x}}_2, ..., \hat{\boldsymbol{x}}_n]^T$, respectively, $\boldsymbol{P}^{(i,j)}$ is the i_{th} row j_{th} column component of interval covariance $\boldsymbol{P}$.

The estimation goal is to have the minimum trace of the error covariance matrix, consequently elements that are not diagonal do not have much interest. However, their overestimated value may result in singularity problems, meaning that candidates of the matrix set have their determinant equal to zero.

Notice that if the matrix $\boldsymbol{P}_{.|.}$ is not an interval matrix, all elements on the diagonal are positive as they represent the variance of each state, thus its trace is positive. In the case of an interval matrix, this particularity no longer holds, due to interval arithmetic [Jaulin 2001d]. Thus a first CSP is introduced which respects the variance definition: each component of the diagonal of $\boldsymbol{P}_{.|.}$, noted $\boldsymbol{P}_{.|.}^{(i,i)}$, is positive, $i = 1, 2..., n$. An operation is given on the interval covariance matrix as follows.

Remark 3.1 *For any $\boldsymbol{P}^{(i,i)}, i = 1, 2..., n$ the value can not be evaluated by $(x_i - \hat{\boldsymbol{x}}_i)(x_i - \hat{\boldsymbol{x}}_i))$; instead $\boldsymbol{P}^{(i,i)}$ should be calculated from $(x_i - \hat{\boldsymbol{x}}_i)^2$ in order to avoid multi-occurrence effect:*

$$\begin{cases} \boldsymbol{P}^{(i,j)} = E\left((x_i - \hat{\boldsymbol{x}}_i)(x_j - \hat{\boldsymbol{x}}_j)\right), i \neq j, \\ \boldsymbol{P}^{(i,i)} = E\left((x_i - \hat{\boldsymbol{x}}_i)^2\right), \\ i = 1, ..., n, j = 1, ..., n. \end{cases} \tag{3.28}$$

.

Thus we obtain a constraint on the value of the variance and consequently on the matrix diagonal and it has the same matrix diagonal particularity as in the punctual case, $tr(\boldsymbol{P}) \geqslant 0$ and $\boldsymbol{P}^{(i,i)} \geqslant 0, i = 1, 2..., n$

As we saw above, the prediction error covariance should also respect the fact that $\hat{\boldsymbol{P}}_{k+1|k}^{i,i} \geqslant 0, i = 1, 2..., n$, which is not guaranteed in interval arithmetic. An extra constraint is therefore added:

$$\begin{cases} \hat{\boldsymbol{P}}_{k+1|k} = \boldsymbol{A}\hat{\boldsymbol{P}}_{k|k}\boldsymbol{A}^T + Q, \\ \mathcal{C}_p : \hat{\boldsymbol{P}}_{k+1|k}^{(i,i)} \geqslant 0, i = 1, 2..., n. \end{cases} \tag{3.29}$$

There exists also other contractors that can contract optimally an interval matrix without loosing any positive semi-definite matrix; for example in [Jaulin 2013], authors showed that the problem of computing the smallest interval sub matrix of a given interval matrix which contains all symmetric positive semi definite matrices is a linear matrix inequality problem, and a convex optimization problem over the cone of positive semi definite matrices, that can be solved in polynomial time. To avoid prohibitive execution times, a simple contractor is implemented.

Correction In the correction phase, the purpose is to find a proper value which can improve the *a priori* state estimate by using the actual output measurement, in order to satisfy the criterion of minimum estimation error covariance. For conventional Kalman filter, the gap between predicted state and real state comes from the statistically known error. In the Interval Kalman filter estimation, not only the noise deviates the state prediction from the real state, but the uncertainties in system parameters and previous estimates also play an

important role. It is difficult to distinguish the two different sources of deviation, as they are correlated.

Assuming the correction equation in the set-membership context, we have:

$$\hat{\boldsymbol{x}}_{k+1|k+1} = \hat{\boldsymbol{x}}_{k+1|k} + K_{k+1}\left(\boldsymbol{y}_{k+1} - \hat{\boldsymbol{y}}_{k+1|k}\right). \tag{3.30}$$

where K_{k+1} represents the gain. The relation between K_{k+1} and K_k is evaluated by:

$$K_{k+1} = \left(A\left(I - K_k C\right)P_{k|k-1}A^T + Q\right)\cdot C^T \cdot \left(C\left(A\left(I - K_k C\right)P_{k|k-1}A^T + Q\right)C^T + R\right)^{-1} \tag{3.31}$$

Intuitively, K_{k+1} aims to bring back the estimate enclosure around the real state while still keeping all the possible values corresponding to uncertainty.

Equations (3.26), (3.27) and (3.30) give the estimation error covariance expression. This manipulation is only valid when the noise is centered gaussian:

$$\hat{\boldsymbol{P}}_{k+1|k+1} = \hat{\boldsymbol{P}}_{k+1|k} - K_{k+1}\boldsymbol{C}\hat{\boldsymbol{P}}_{k+1|k} - \hat{\boldsymbol{P}}_{k+1|k}(\boldsymbol{C}^T K_{k+1}^T + K_{k+1}\left(\boldsymbol{C}\hat{\boldsymbol{P}}_{k+1|k}(\boldsymbol{C}^T + R\right)K_{k+1}^T. \tag{3.32}$$

We find K_{k+1} that minimizes $trace(\hat{\boldsymbol{P}}_{k+1|k+1})$. The reason why using $trace(\cdot)$ is that state variance at matrix diagonal is in fact the value that indicates the estimation error:

$$\begin{aligned}
\frac{\partial trace(\hat{\boldsymbol{P}}_{k+1|k+1})}{\partial K_{k+1}} &= -2\hat{\boldsymbol{P}}_{k+1|k}\boldsymbol{C}^T s + 2K_{k+1}\left(\boldsymbol{C}\hat{\boldsymbol{P}}_{k+1|k}\boldsymbol{C}^T + R\right).\\
\frac{\partial^2 trace(\hat{\boldsymbol{P}}_{k+1|k+1})}{\partial K_{k+1}\partial K_{k+1}^T} &= 2\left(\boldsymbol{C}\hat{\boldsymbol{P}}_{k+1|k}\boldsymbol{C}^T + R\right).
\end{aligned}$$

The second derivative is always positive in the conventional Kalman filter, which guarantees the existence of a minimization solution. We can prove that in the set-membership context it is still valid if and only if the second derivative, which is a box, stays positive. This result set includes all possible vertices for any possible coefficient combinations within bounded uncertainty. If we inject the obtained set of vertices into the original function, we have a set of minimum function values guaranteed in the solution, as we expected.

In this case, K_{k+1} turns out to be an interval matrix, which we can note as $\boldsymbol{K}_{k+1} \in \mathbb{IR}^{n\times m}$, its value can be determined from the first order derivative which must be zero:

$$\boldsymbol{K}_{k+1} = \hat{\boldsymbol{P}}_{k+1|k}\boldsymbol{C}^T\left(\boldsymbol{C}\hat{\boldsymbol{P}}_{k+1|k}\boldsymbol{C}^T + R\right)^{-1}. \tag{3.33}$$

Equations (3.32) and (3.33) give the estimation error covariance expression:

$$\hat{\boldsymbol{P}}_{k+1|k+1} = (I_n - \boldsymbol{K}_{k+1}\boldsymbol{C})\hat{\boldsymbol{P}}_{k+1|k}. \tag{3.34}$$

The interval matrix gain $\boldsymbol{K}_{k+1}$ in Equation (3.32) produces a set of estimation error covariances which contain (but not only) the minimum values of all possible uncertainties.

In comparison, the conventional Kalman gain yields the only one minimum estimation covariance.

The Equation (3.33) has a matrix inverse operation, which means the following property should be fulfilled:

$$0 \notin det\left(\boldsymbol{C}\hat{\boldsymbol{P}}_{k+1|k}\boldsymbol{C}^T + R\right).$$

$\boldsymbol{C}$ occurs in the formula several times, the constraint applied on Equation (3.29) should also be used on $\boldsymbol{C}\hat{\boldsymbol{P}}_{k+1|k}\boldsymbol{C}^T$.

By using the updated state estimate and estimation error covariance, we can proceed to the next iteration; the loop is then completed.

3.6.2 Algorithm loop

Equations (3.24), (3.25), (3.33), (3.34) and (3.30) constitute a discrete interval Kalman filtering algorithm that we note as **iIKF**.

Initialization:

$$\begin{aligned} \hat{\boldsymbol{x}}_{0|0} &= x_0, where \quad x_0 \sim \mathrm{N}(m_0, P_0), \\ \hat{\boldsymbol{P}}_{0|0} &= P_0, \end{aligned} \tag{3.35}$$

prediction:

$$\begin{aligned} \hat{\boldsymbol{x}}_{k+1|k} &= \boldsymbol{A}\hat{\boldsymbol{x}}_{k|k} + \boldsymbol{B}\boldsymbol{u}_k, \\ \hat{\boldsymbol{P}}_{k+1|k} &= \boldsymbol{A}\hat{\boldsymbol{P}}_{k|k}\boldsymbol{A}^T + Q, \\ \mathcal{C}_p &: \hat{\boldsymbol{P}}^{(i,i)}_{k+1|k} \geqslant 0, i = 1, 2..., n, \\ k &= 0, 1, ... \end{aligned}$$

correction:

$$\begin{aligned} \boldsymbol{K}_{k+1} &= \hat{\boldsymbol{P}}_{k+1|k}\boldsymbol{C}^T\left(\boldsymbol{C}\hat{\boldsymbol{P}}_{k+1|k}\boldsymbol{C}^T + R\right)^{-1}, \\ \mathcal{C}_p &: [\boldsymbol{C}\hat{\boldsymbol{P}}_{k+1|k}\boldsymbol{C}^T]^{(i,i)} \geqslant 0, i = 1, 2..., n, \\ \hat{\boldsymbol{P}}_{k+1|k+1} &= (I_n - \boldsymbol{K}_{k+1}\boldsymbol{C})\hat{\boldsymbol{P}}_{k+1|k}, \\ \hat{\boldsymbol{x}}_{k+1|k+1} &= \hat{\boldsymbol{x}}_{k+1|k} + \boldsymbol{K}_{k+1}\left(\boldsymbol{y}_{k+1} - \hat{\boldsymbol{y}}_{k+1|k}\right), \\ k &= 0, 1, ... \end{aligned}$$

3.6.3 Over estimation control

3.6.3.1 Gain value propagation

The interval matrix $\boldsymbol{C}\boldsymbol{P}\boldsymbol{C}^T + R$ in Equation (3.33) may have a singularity and the inverse algebraic operation is difficult. Besides, the interval matrix inverse is obtained by

approximation algorithms like in [Rohn 1993] which generally produces over estimation.

We suggest an approach which uses set inversion, in particular the algorithm SIVIA. Applying The idea in Section 2.4.4, we replace the interval matrix inversion problem by a list of constraint propagation problems. The Equation (3.33) is rewritten without the inverse operation:

$$\boldsymbol{K}_{k+1}\left(\boldsymbol{C}\hat{\boldsymbol{P}}_{k+1|k}\boldsymbol{C}^T + R\right) = \hat{\boldsymbol{P}}_{k+1|k}\boldsymbol{C}^T.$$

We define $\boldsymbol{S}_{k+1} = \boldsymbol{C}\hat{\boldsymbol{P}}_{k+1|k}\boldsymbol{C}^T + R$, $\boldsymbol{T}_{k+1} = \hat{\boldsymbol{P}}_{k+1|k}\boldsymbol{C}^T$,

$$\boldsymbol{K}_{k+1}\boldsymbol{S}_{k+1} = \boldsymbol{T}_{k+1}. \tag{3.36}$$

where $\boldsymbol{K}_{k+1} \in \mathbb{IR}^{n\times m}$, $\boldsymbol{S}_{k+1} \in \mathbb{IR}^{m\times m}$, $\boldsymbol{T}_{k+1} \in \mathbb{IR}^{n\times m}$.

Every component in matrix $\boldsymbol{K}$ is considered separately and the search space is the Cartesian product of each component:

$$\boldsymbol{K}_{k+1}^{(1,1)} \times \boldsymbol{K}_{k+1}^{(1,2)} \times ... \times \boldsymbol{K}_{k+1}^{(n,m)}.$$

This search space is then bisected and tested under SIVIA properly adapted to matrix operations. The result is a set of small boxes that satisfy Equation (3.36).

The initial search space of $\boldsymbol{K}_{k+1}$ can be obtained by several techniques. The Equation (3.33) can still be used to give a large approximation of the gain, but in that case, matrix singularity problems remain; another approach is to use constraint propagation (such as Forward-backward algorithm) in order to obtain a reasonable initial sub space from a very large search space.

Each box can establish a "small acceptable gain"; the set of boxes is then injected into the correction step to update covariance matrix and state estimate vector. The final result is the hull of all the covariance matrices and state estimate vectors corrected by each small gain.

3.6.3.2 Constraint Propagation

The traditional algorithms for constraint satisfaction problem can also be applied in interval analysis. In [Jaulin 1993], these algorithms are extended to the interval arithmetic for linear and nonlinear system.

Among these extensions, the forward-backward algorithm is adapted to our implementation. The principle is to decompose the constraint equation $f(\boldsymbol{x_1}, ..., \boldsymbol{x_n}) = 0$ in a sequence of elementary operations of primitive functions like $\{+, -, \times, /\}$ to construct a list of primitive constraints ([Lhomme 1996]). For example, equation:

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + K_{k+1}(y_{k+1} - C\hat{x}_{k+1|k}),$$

can be decomposed into a set of following primitive constraints:

$$\begin{cases} a_1 = C\hat{x}_{k+1|k}, \\ a_2 = y_{k+1} - a_1, \\ a_3 = K_{k+1}a_2, \\ \hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + a_3. \end{cases}$$

We seek to contract $\{\hat{\boldsymbol{x}}_{k+1|k+1}, \hat{\boldsymbol{x}}_{k+1|k}, \boldsymbol{K}_{k+1}\}$ by the propagation of constraints without changing $\{\boldsymbol{C}, \boldsymbol{y}_{k+1}\}$ as they are considered as inputs.

The advantage of implementing CSP in interval analysis is to accelerate the computation and to find a less overestimated result. In [Lhomme 1996, Ramdani 2010, Jaulin 2001b], this advantage has been presented.

3.6.3.3 Interval intersection rule

As the associative law is no longer valid in interval arithmetic, we must redefine the product of interval matrices to control overestimation like in [Li 2008], which yields the principle of the interval intersection rule:

$$\prod_{i=1}^{n} \boldsymbol{M}_i \triangleq \left[(\prod_{i=1}^{n-1} \boldsymbol{M}_i) \cdot \boldsymbol{M}_n \right] \cap \left[\boldsymbol{M}_1 \cdot (\prod_{i=1}^{n-1} \boldsymbol{M}_{i+1}) \right]. \tag{3.37}$$

This definition is applied systematically when there is multiplication of 3 or more interval matrices in our work, which improves the result of product.

3.6.3.4 Adaptive calibration

As a last operation to control overestimation, a calibration can be implemented to reset the iteration for limiting the divergence. The idea appeared in [Li 2008]. In our approach, the calibration can be trigged when the interval matrix to-be-inverse is not regular:

$$\hat{\boldsymbol{x}}_k \triangleq \boldsymbol{C}_k, \ \boldsymbol{P}_k = \boldsymbol{P}_0. \tag{3.38}$$

where $\boldsymbol{C}_k$ is supposed to be a smaller box than $\hat{\boldsymbol{x}}_k$ and is approaching or containing the real state with high confidence. In practice, one can use the value estimated from the nominal system where the parameters are middle (or nominal) value of interval matrices.

At other times, a threshold can be set to verify if the calibration should take place. The threshold can be obtained by the following equation based on Equation (**??**):

$$\begin{aligned} x_{k+1} = &\sum_{i=1}^{k+1} A^{k+1-i} w_{i-1} + \sum_{n=1}^{k} \left((A + \Delta A)^n - A^n \right) w_{k-n} + \\ &(A + \Delta A)^{k+1} x_0 + \sum_{n=1}^{k+1} \left((A + \Delta A)^{n-1} (B + \Delta B) - A^{n-1} B \right) u_{k+1-n} + \sum_{i=1}^{k+1} A^{k+1-i} B u_{i-1}. \end{aligned}$$

In case of a command free system, this equation can be simplified to:

$$x_{k+1} = \sum_{i=1}^{k+1} A^{k+1-i} w_{i-1} + \sum_{n=1}^{k} \left((A + \Delta A)^n - A^n\right) w_{k-n} + (A + \Delta A)^{k+1} x_0.$$

In practice, we replace the noise element w in the equation by its standard derivation interval $[-3\sqrt{q}, 3\sqrt{q}]$ to approximate 99.7% cases of generated noise, where q is the variance vector of noise sequence w. This approximation gives an estimated state $\boldsymbol{x}^*$ from the initial condition:

$$\boldsymbol{x}^*_{k+1} = \boldsymbol{A}^{k+1} x_0 + \sum_{i=1}^{k+1} \boldsymbol{A}^{k+1-i} [-3\sqrt{q}, 3\sqrt{q}]. \tag{3.39}$$

The threshold for calibration is then set to $\boldsymbol{x}^*_{k+1}$, that means, if:

$$w(\boldsymbol{x}(i)_{k+1|k+1}) \geqslant w(\boldsymbol{x}(i)^*_{k+1}), i = 1, ..., n,$$

the calibration will take place. It is important to notice that after the calibration, the current state is set to new x_0 by Equation (A.5).

In the following section, we implement the different filters on a simple example and compare the results with the original Interval Kalman filtering sIKF from [Chen 1997].

3.7 Case study

Application to case study is the best way to test the efficiency of new methods. In this section, two examples are used to compare side by side all methods mentioned in previous sections. In order to evaluate the performance of each method, three indexes are introduced: **N** , **O** and **D**. N is the number of calibration times, O is the number of times that estimate envelope does not contain real state, and D the norm describing the distance of interval estimate bounds and the true value. D is given by:

$$\begin{cases} D = \dfrac{\sqrt{\sum_{k=1}^{K} d(\hat{\boldsymbol{x}}_k, x_k)^T d(\hat{\boldsymbol{x}}_k, x_k)}}{\sqrt{\sum_{k=1}^{K} x_k^T x_k}}, \\ d(\hat{\boldsymbol{x}}_k, x_k) = (sup(\hat{\boldsymbol{x}}_k) - x_k) + (inf(\hat{\boldsymbol{x}}_k) - x_k). \end{cases} \tag{3.40}$$

where K represents the maximum iteration number.D is the less the better, which implies the algorithm is less divergent, O is also the less the better. However, for D, though it is generally the less the better, it is also influenced by the value of N, when the calibration happens more, the distance of estimated bound can also be smaller. Besides, method IRLS requires a forget factor λ, and the algorithm SIVIA requires a user-specified precision threshold ε. These factors are also investigated in our work. Lastly t is the execution time.

3.7.1 Example 1: Comparison of iIKF, sIKF and IRLS with different forget factors

Firstly, we apply the original filter IKF, sIKF and IRLS on the example originally presented in [Chen 1997] which describes a simple radar tracking system.

$$\begin{cases} x_{k+1} = \begin{bmatrix} 1 & \boldsymbol{h} \\ 0 & 1 \end{bmatrix} x_k + w_k, \\ y_k = \begin{bmatrix} 1 & 0 \end{bmatrix} x_k + v_k, \ k = 0, 1, 2, ... \end{cases}$$

where $\boldsymbol{h} = h_0 + \triangle h$, with $h_0 = 0.01$ et $\triangle h = [-0.001, 0.001]$, w_k and v_k are known noises in the Equation (3.1).

$$E\{x_0\} = \begin{bmatrix} x_{01} \\ x_{02} \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix},$$
$$Cov\{x_0\} = \begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix} = \begin{bmatrix} 0.5 & 0.0 \\ 0.0 & 0.5 \end{bmatrix},$$
$$Q = \begin{bmatrix} q & 0 \\ 0 & q \end{bmatrix} = \begin{bmatrix} 0.1 & 0.0 \\ 0.0 & 0.1 \end{bmatrix}, \ R = r = 0.1.$$

For IRLS, the forget factor $\lambda = 0.3, 0.5, 0.7$ *and* 0.9 are investigated, the calibration threshold is fixed to 1 as the largest width of the results for x_1.

Using Matlab^{TM} the results are written in table 3.2.

Filtre	λ	**N**	**O**	**D**	**t**
sIKF	-	29	177	0.427	76
iIKF	-	0	0	0.928	105
IRLS	0.3	51	161	0.552	137.58
	0.5	37	147	0.535	133.03
	0.7	25	159	0.608	135.82
	0.9	14	191	0.797	136.88

Table 3.2: Comparison of the IKF, sIKF and IRLS

These approaches give satisfactory results if the noises are known and independent, which is the precondition of the classical Kalman filter.

By comparing the obtained estimate of iIKF, sIKF and IRLS with their conventional origins, it is noted that the estimated set bounds $\hat{\boldsymbol{x}}_k$ envelope the estimated value obtained in conventional point based approaches, however in sIKF and also in IRLS, there are sometimes the bounds do not envelope the real states, due to the choice of replacing the interval matrix by its upper bound.

The behaviour of filter IRLS varies with different forget factors, which is a compromise between the calibration number N and the estimation error D.(see Figure 3.5 et Figure 3.6)

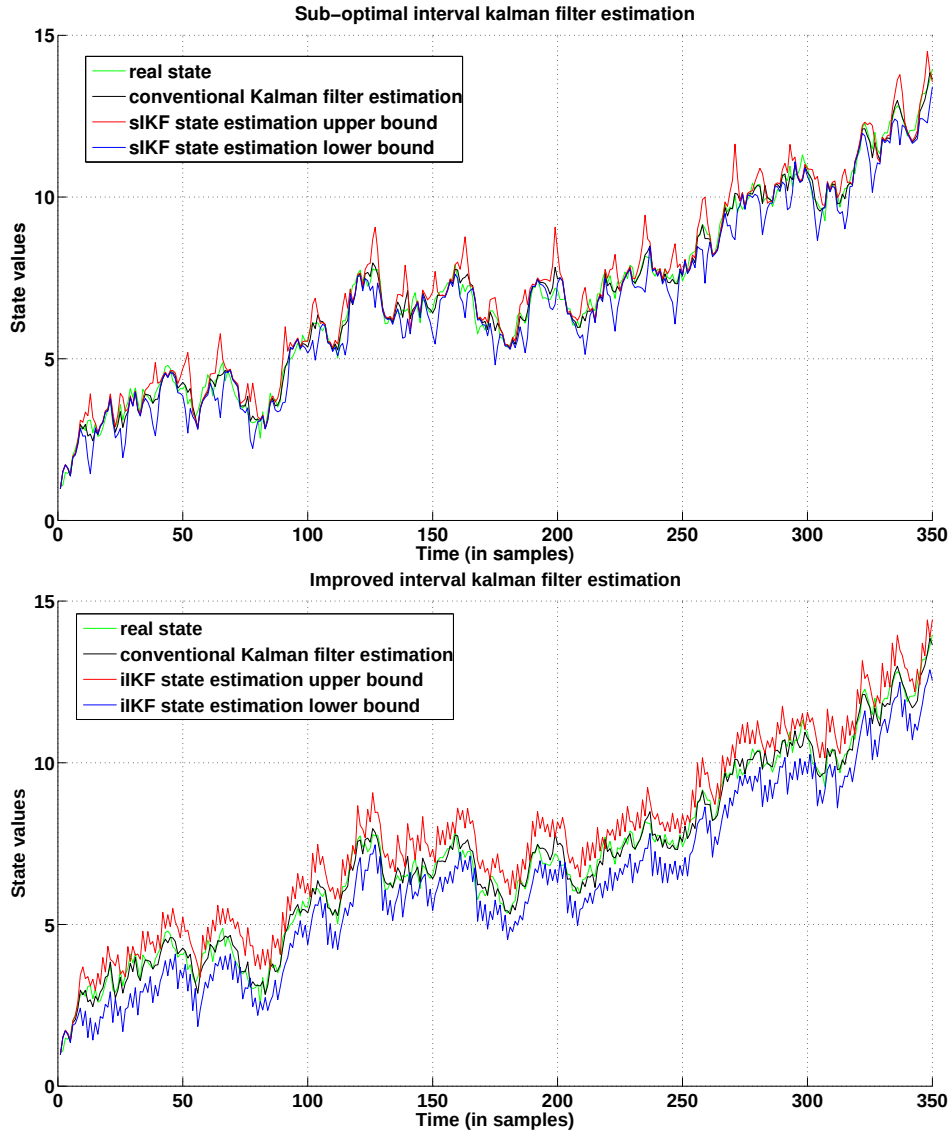

Figure 3.5: Result of estimation from sIKF (up) / Result of estimation from iIKF (down).

Our filter iIKF has better performance than the IRLS by comparing all the variables D, O and t, under the condition that the noises are well known. Once the characteristics of noises are not as precise as expected,the simulation of iIKF will encounter an important degeneration, while the IRLS resists (result not shown in the test).

By comparing the iIKF and sIKF, we notice an obvious improvement in terms of N and O, although the execution time is longer in iIKF. The reason is that in iIKF, the SIVIA based filter gain and constraint propagation has a better control on overestimation which limits the calibration time and also improves the enclosure of estimation result, but these operations demand more compute resource and time.

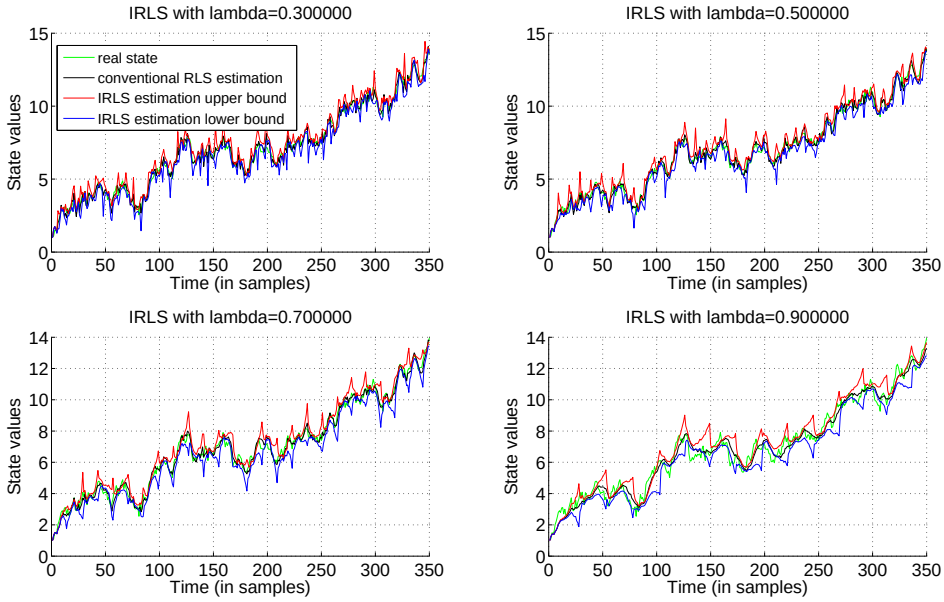

Figure 3.6: Results of IRLS with different forget factors

3.7.2 Example 2: Comparison of the sIKF and the iIKF with different SIVIA ε factor

Another demonstration is established on a rewritten example proposed in [Chen 1997], originally appeared in [Zhang 1995], in order to compare sIKF and iIKF with different set of ε.

This is an interesting example as it is allowed that the noise covariance matrices Q and R have perturbations $\triangle Q$ and $\triangle R$, respectively:

$$\begin{cases} x_{k+1} = \boldsymbol{A}x_k + w_k, \\ y_k = \boldsymbol{C}x_k + v_k, k = 0, 1, 2, ... \end{cases}$$

where w_k et v_k are noises modelled by known Gaussian distribution, $\boldsymbol{A} = A + \triangle A, \boldsymbol{C} = C + \triangle C$,

$$A = \begin{bmatrix} 0.4 & 0.1 \\ -0.1 & 0.2 \end{bmatrix}, C = \begin{bmatrix} 0 & 1 \end{bmatrix}, Q = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}, \ R = 1.$$

Filter	ε	N	O	D	t
IKF	-	18	14	18.4770	0.93s
sIKF	-	0	56	0.85	0.75s
iIKF	1	0	0	1.5382	11s
	0.2	0	0	1.5098	55s
	0.05	0	0	1.5036	791s

Table 3.3: Comparison between the interval Kalman filter (IKF) and improved interval Kalman filter (iIKF) with different bisection factors ε.

The bounded perturbations and initial conditions are:

$$\triangle A = \begin{bmatrix} [-0.1, 0.1] & [-0.15, 0.15] \\ 0 & [-0.25, 0.25] \end{bmatrix}, \triangle C = \begin{bmatrix} 0 & [-0.1, 0.1] \end{bmatrix},$$
$$\triangle Q = \begin{bmatrix} [-2, 2] & 0 \\ 0 & [-2, 2] \end{bmatrix}, \triangle R = [-0.9, 1.1],$$
$$E\{x_0\} = \begin{bmatrix} x_{01} \\ x_{02} \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix},$$
$$Cov\{x_0\} = \begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix} = \begin{bmatrix} 0.5 & 0.0 \\ 0.0 & 0.5 \end{bmatrix}.$$

The covariance matrices are time-invariant and considered without uncertainty. The goal is to determine the additive fault on the measurement as soon as possible.

The simulation is done on the time stage [0,100]. By using Matlab^{TM} and INTLAB^{TM} toolbox ([Rump 1999]),we obtained results in table 3.3. Notice that $\varepsilon = 1$ means there is no gain value propagation. These results are consistent with that shown in Figures 3.7 (left) and 3.7 (right).

We can see that the original IKF has the largest D, meaning without over-estimate control causes divergence in implementations, while sIKF has the minimum value of D which corresponds to narrower boundaries of interval estimates. But since it replaces the uncertainty matrix to-be-inverse by its upper bound, certain solutions are lost, which leads to a largest value of O: more than half time the real state is outside the estimate envelope.

By using iIKF, D is larger than sIKF, because it keeps all bounded uncertainties in its loop. We notice that the real state and optimal solution from conventional Kalman filter are both entirely contained in boundary. The gain value propagated from SIVIA actually refines the interval value, but it is time consuming as predefined precision increases. Compare to original IKF, our approach prevents unnecessary recalibration due to the divergent interval operation; compare to sIKF, the new approach keeps all the solutions derived from bounded error uncertainty.

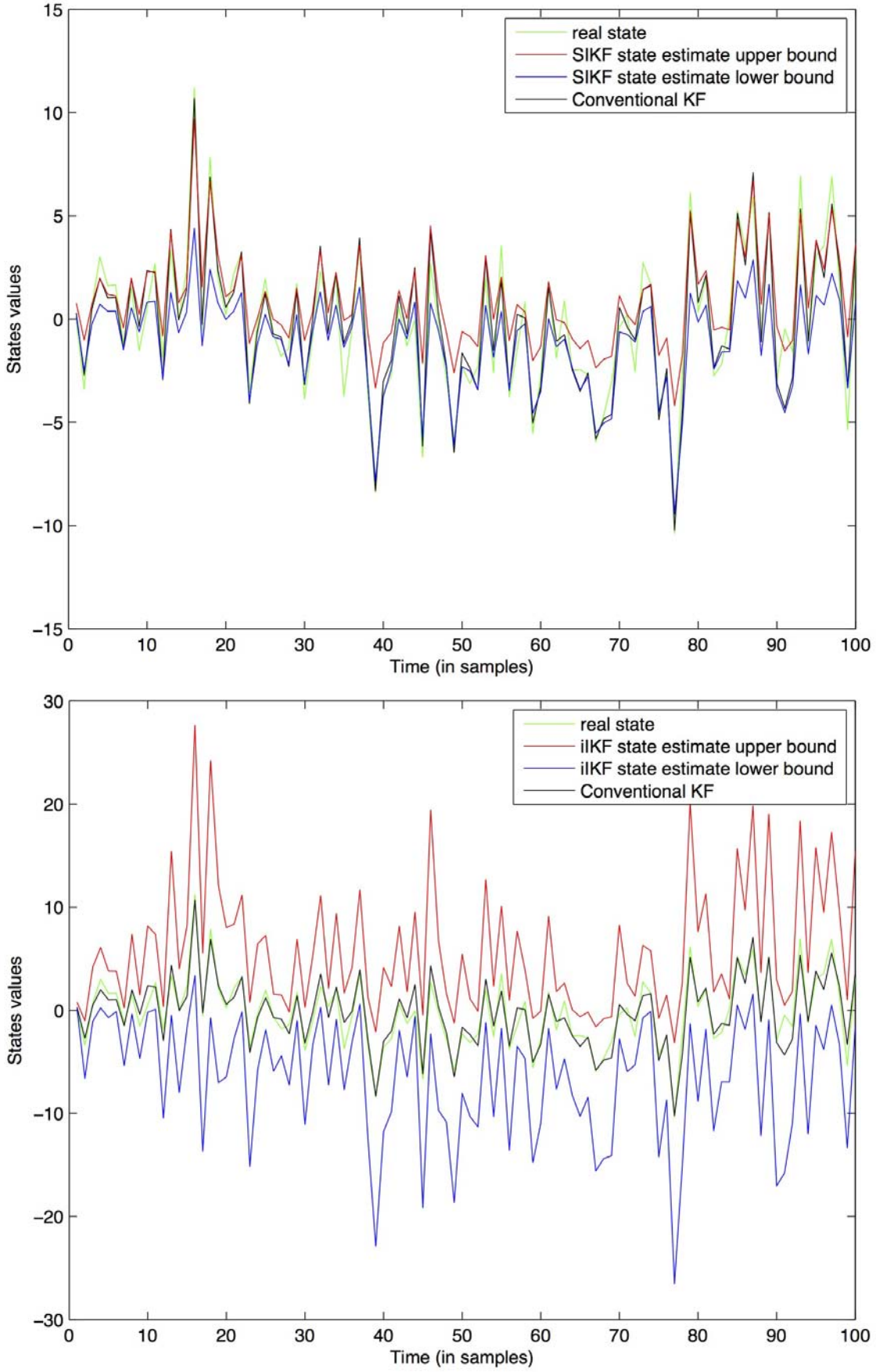

Figure 3.7: Simulation results from sIKF (up) / Simulation results from iIKF with $\varepsilon = 0.05$ (down).

3.8 Conclusions

In this chapter, our main contribution consists in proposing an alternative approach to solve the interval matrix inversion problem without loss of solutions while controlling the inherent pessimism of interval calculus. The objective is to combine the natural statistical uncertainty and set-membership uncertainty in system called "Uncertain Stochastic System". Several techniques have been implemented to limit overestimation effects propagating within the interval Kalman filter recursive structure, in particular constraints on the interval covariance matrix are added. In particular the gain of the filter is obtained by a calculus based on the set inversion algorithm SIVIA. This method can be extended to all linear system for solving inversion problem. The results have shown that the iIKF provides guaranteed results while controlling the pessimism of interval calculus.

The improved IKF can be applied to obtain estimation of all the Kalman filter by taking account non linear parametric uncertainty, which will be presented in the next chapter.

Chapter IV
Adaptive threshold based fault detection

CHAPTER 4

Adaptive threshold based fault detection

Contents

4.1 Introduction

Fault detection and isolation play a crucial role in enhancing the safety and reliability, and reducing the operating cost of any related systems. However, achieving the FDI task with high reliability is a challenging problem especially when the circumstance is complicated and different uncertainties are present. The model based fault detection consists in comparing the behaviours of the process and its model when both are fed with the same inputs. As generally the process state is unknown, this comparison is achieved from the process output and the model output, where the latter is reconstructed from a state estimator. The extensions of the Kalman filter to deal with disturbances, noises and parameter uncertainties at the same time was presented in the previous chapters.

In practice, divergence between the measured and predicted output arising from noises are corrected by the filtering. But it can also come from faults. The purpose of the work is to detect the abnormalities as fast as possible by taking account the modeled uncertainties.

The diagnostic problem consists in supervising the system behaviour and alerting as early as possible when a fault appears, then in identifying the faulty component. We

present not only the standard method of fault detection using the state estimation from the Kalman filter, but also the set-membership estimation based fault detection; in the following section we use our estimation approach, the iIKF, to achieve fault detection. The main challenge is the residual generator, which is investigated on several sensor faults in two case studies.

4.2 Classical fault detection

Different types of faults exist in a system, such as actuator faults, sensor faults and component faults. Each type of fault is modelled differently.

In our work, only sensor faults are considered, the following cases are considered:

The blocking value - In such case, the measured outputs y_m stay at some constant value starting from instant k_f:

$$y_m(k) = const, \forall k \geqslant k_f, \tag{4.1}$$

where *const* represents constant vector and k_f is the time instant at which the fault occurs.

The zero value - The measured output y_m is zero after the fault happens:

$$y_m(k) = 0, \forall k > k_f. \tag{4.2}$$

Additive fault - This fault is trigged when there is a bias or calibration problem. The measured output is the theoretical value plus an additive term, the theoretical value $y_m^*(k)$ plus an additive term f_a:

$$y_m(k) = y_m^*(k) + f_a, \forall k > k_f. \tag{4.3}$$

Multiplicative fault - In this case, a multiplicative factor changes the nominal value:

$$y_m(k) = f_m y_m^*(k), \forall k > k_f, \tag{4.4}$$

where f_m is the multiplicative fault factor.

4.2.1 Fault detection with conventional Kalman filter

The Kalman filter can be used for fault detection. Under the bounded situation where there is no set-valued uncertainty, the Kalman filter provides an optimal state estimate. The state estimate can then be used to calculate the model output and the result is compared to the output measurement.

The residual at time $k+1$ can be obtained by the following equation:

$$r_{k+1} = y_{k+1} - \hat{y}_{k+1|k}, \tag{4.5}$$

where y_{k+1} represents the measured output at time $k+1$, and $\hat{y}_{k+1|k}$ represents the *a priori* estimated output.

A distance measure can be used to evaluate the deviation, for example, one may use directly the residual:

$$D_{k+1} = r_{k+1}, \tag{4.6}$$

or one may also use a moving average filter distance measure to give an adjustable relevance for the last samples and present sample, depending on whether the system designer considers the past estimation history less relevant. This distance measure is defined as:

$$D_{k+1} = (1-\lambda)D_k + \lambda r_{k+1}, \tag{4.7}$$

where λ varies from 0 to 1. A larger value of λ indicates that the present estimation has more impact on the distance measure D.

There exists a "normal" difference between the predicted output and the measured output, which is due to the output noise. When the measured output is not faulty and without measurement noise, the residual is reduced to zero. When a fault occurs, the residual vector is no longer null and at least one of its components indicates the fault. When noise is present, the residual should be evaluated statistically. A classical approach is to apply a statistical test to the distance measure using a threshold as an estimate of the standard deviation from the predicted output at each time, known as the 3σ test [Isermann 1997a]. The area between two deviation curves includes 99.7% of the possible values deviated from the theoretical value due to the noise. The measured output that exceeds the range of such interval can be marked as a fault. Illustrated in figure 4.1, a sensor fault occurs at the sample time $k = 50$. It can be detected by the Kalman filter method by testing if the measured output exceeds the thresholds.

Another approach is to check the generalized variance [1] of the Kalman filter innovation sequence [Hajiyev 1999]. In the Kalman filter, there is an innovation sequence following the normal distribution, which has a zero mean and a unit covariance matrix when normalized [Hajiyev 1998]. In fact, for sensor faults, the mean, the covariance matrix are altered from their "normal values". However this approach is not investigated in this work.

The applications of the Kalman filter and its extensions on non-linear systems for fault detection can be found in [Mosallaei 2008].

4.2.2 Fault detection with bounded noise or disturbances

Residual generation for uncertain models is different. In many cases, the interval based approaches for fault detection have different assumptions on noise. In these approaches such as [Ingimundarson 2009], the only information on noise and disturbances is that they

1. generalized variance is the determinant of correlation matrix

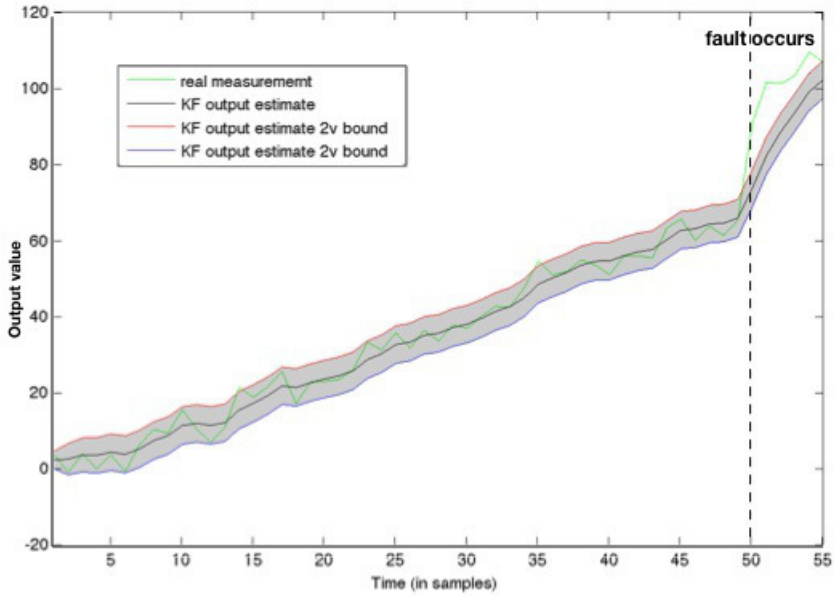

Figure 4.1: Fault detection by using Kalman filter

are knowingly bounded too, which gives a system modeled by the interval version of Equation (1.8):

$$\begin{cases} x_{k+1} = f(x_k, u_k, \boldsymbol{\theta}) + \boldsymbol{w}_k, \\ y_k = g(x_k, u_k, \boldsymbol{\gamma}) + \boldsymbol{v}_k, k = 0, 1, 2, ... \end{cases}$$

In this equation, $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are no longer Gaussian but bounded boxes, the parameters are also uncertain and bounded. Although these are still unknown sequences, the fact that the disturbances belong to known compact sets enables fault detection by testing the consistency of the estimated uncertain state or output with the measured ones.

The process involves the state estimation and the measured output inverse; an admissible domain can be obtained by the intersection of the domain obtained using the model equation and the measurement equation. If the admissible domain is empty, an inconsistency between the system model and the measured output is detected.

We should notice that the domain evaluated from measured outputs is defined by:

$$D_{y,k+1} = \{y \mid inf(\boldsymbol{v}_{k+1}) < y_{k+1} - y_{m,k+1} < sup(\boldsymbol{v}_{k+1})\}. \tag{4.8}$$

The estimation domain is given by:

$$D_{\hat{y},k+1} = \{\hat{y} \mid \hat{y} = g(\hat{x}_{k+1}, u_{k+1}, \boldsymbol{\gamma}_{k+1}) + \boldsymbol{v}_{k+1}, \hat{x}_{k+1} = f(\hat{x}_k, u_{k+1}, \boldsymbol{\theta}_{k+1}) + \boldsymbol{w}_{k+1}\}. \tag{4.9}$$

The residual can be expressed as:

$$r_{k+1} = D_{y,k+1} \cap D_{\hat{y},k+1} \tag{4.10}$$

This domain can be achieved in the case where the disturbance can be modelled as a known bounded set. Formally, a fault is detected when:

$$r_{k+1} = \emptyset. \tag{4.11}$$

The intersection of two domains is not necessarily convex, which may result in an important computational load. Sometimes, the intersection domain is approximated by an overestimated simpler form. The overestimation may imply no detection of some faults. The difficulty in reality is to find a compromise between the complexity of the state domain or the output domain and the tolerable rate of the missed alarms.

A similar method is adopted in different applications, such as [Kesavan 2001], which uses the parallelotope covering the feasible region of the initial state propagated by using the 'Recursive Optimal Bounding Parallelotope' methodology.

4.2.3 Diagnosis

The previous procedures for fault detection make it possible to detect inconsistency in the data. In the point-wise case, under the context of stochastic noise, the Kalman filter uses the standard deviation to test if the measured output is statistically consistent with the predicted output defined by the healthy model; in the set-valued case, under the assumption of bounded disturbances and uncertain parameters, interval arithmetic and the intersection operation are used to test the consistency between the measured outputs and the output estimated from the model.

But the diagnosis does not stop here. The complete procedure includes fault isolation. A solution consists in computing the state estimate using only part of the output measurements: one may design a bank of dedicated observers where each observer is associated with specific component [Xue 2007]. Each observer or a group of observers represents an indicator for one possible fault. Because the principles of fault isolation remain the same as those of fault detection, with several instead of one observer, we focus the work presented in the rest of the chapter on fault detection.

4.3 Fault detection in Uncertain Stochastic System

In our work, the bounded parameters and statistical noises are both included in the system, so the fault detection approach should be reconsidered.

In the case of discrete time linear models with bounded uncertainties on parameters and Gaussian output noise, the residual analysis is more complex than in the standard cases presented in sections 4.2.1 and 4.2.2.

4.3.1 Adaptive threshold

Let us consider the same system model (3.2) as chapter 3:

$$\begin{cases} x_{k+1} = Ax_k + Bu_k + w_k, \\ y_k = Cx_k + Du_k + v_k, \end{cases}$$

where $A \in \boldsymbol{A}$, $B \in \boldsymbol{B}$, $C \in \boldsymbol{C}$, $D \in \boldsymbol{D}$. The matrices A, B, C, D are "punctual", while $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D}$ are interval matrices. The state equation and output equation at time k have centered Gaussian noises w and v.

In the proposed work, we consider additive faults on sensors and adopt the single fault assumption i.e. only one fault is present at a time. The faults can be modelled by modifying the observation equation of (3.2) as follows :

$$y_k = Cx_k + Du_k + v_k + f_a, \forall k > k_f, \tag{4.12}$$

where f_a represents an additive fault vector and k_f is the time when the fault occurs.

We can also consider the case of a non-persistent fault, which can be interpreted as:

$$y_k = Cx_k + Du_k + v_k + f_a, \forall k_r > k > k_f, \tag{4.13}$$

where k_r is the time when the system restores to healthy.

The state observer is based on the uncertain stochastic model:

$$\begin{cases} \boldsymbol{x}_{k+1} = \boldsymbol{A}x_k + \boldsymbol{B}u_k + w_k, \\ \boldsymbol{y}_k = \boldsymbol{C}x_k + \boldsymbol{D}u_k + v_k. \end{cases}$$

The state estimate $\hat{\boldsymbol{x}}$ is obtained by the iIKF, including the *a priori* estimate $\hat{\boldsymbol{x}}_{k+1|k}$ and the *a posteriori* estimate $\hat{\boldsymbol{x}}_{k+1|k+1}$.

$$\begin{cases} \hat{\boldsymbol{x}}_{k+1|k} = \boldsymbol{A}\hat{\boldsymbol{x}}_{k|k} + \boldsymbol{B}u_k, \\ \hat{\boldsymbol{x}}_{k+1|k+1} = \hat{\boldsymbol{x}}_{k+1|k} + \boldsymbol{K}_{k+1}(y_{m,k+1} - \hat{\boldsymbol{y}}_{k+1|k}), \end{cases}$$

The *a posteriori* state estimate $\hat{\boldsymbol{x}}_{k+1|k+1}$ is corrected by the Kalman gain, which is evaluated based on the updated output measurement $y_{m,k+1}$, thus it is highly possible that this estimate value follows the faulty measurement when there exist sensor problem. That is why in following work the *a posteriori* state estimate is not used in the fault detection process. On the other hand, the *a priori* state estimate can be used for fault detection if we carefully choose the strategy, which is presented in the next section.

The *a priori* state estimate can be used to obtain the output estimate $\hat{\boldsymbol{y}}_{k+1|k}$ without correction from Kalman filter:

$$\hat{\boldsymbol{y}}_{k+1|k} = \boldsymbol{C}\hat{\boldsymbol{x}}_{k+1|k} + \boldsymbol{D}u_{k+1}. \tag{4.14}$$

Confidence intervals for state and output estimates In the conventional approach, a confidence interval at 99.7% on the i_{th} component of $\hat{x}_{k+1|k}$ ($i = 1, ..., n$) can be used for determining threshold on *a priori* states by using the previous *a priori* state covariance $P_{k|k}$:

$$I^i_{\hat{x},k+1|k} = \left[\hat{x}^i_{k+1|k} - q \times \sqrt{P^{(i,i)}_{k+1|k}},\ \hat{x}^i_{k+1|k} + q \times \sqrt{P^{(i,i)}_{k+1|k}}\right], \tag{4.15}$$

where $\hat{x}^i_{k+1|k}$ is the i_{th} component of the *a priori* state estimate $\hat{x}_{k+1|k}$, $P^{(i,i)}_{k|k}$ is the i_{th} component of diagonal elements of state error covariance matrix $P_{k+1|k}$ and $q = 3$ (to obtain a confidence interval at 99.7%). Thus the probability of all values lying within the thresholds is higher or equal to 99.7%.

This threshold definition can be extended to the set-membership context by using the interval expectation and the covariance matrix obtained by the iIKF. Thus $\boldsymbol{I}^i_{\hat{\boldsymbol{x}},k+1|k}$ is given, for $i = 1, ..., n$, by:

$$\boldsymbol{I}^i_{\hat{\boldsymbol{x}},k+1|k} = \left[\underline{\left(\hat{\boldsymbol{x}}^i_{k+1|k} - q \times \sqrt{\boldsymbol{P}^{(i,i)}_{k+1|k}}\right)},\ \overline{\left(\hat{\boldsymbol{x}}^i_{k+1|k} + q \times \sqrt{\boldsymbol{P}^{(i,i)}_{k+1|k}}\right)}\right], \tag{4.16}$$

where $\hat{\boldsymbol{x}}^i_{k+1|k}$ is the i_{th} component of $\hat{\boldsymbol{y}}_{k+1|k}$ and $\boldsymbol{\sigma}^i_{k+1}$ represents the standard deviation of $\hat{\boldsymbol{y}}^i_{k+1|k}$.

Also in the conventional approach, a confidence interval at 99.7% on the i_{th} component of $\hat{y}_{k+1|k}$ ($i = 1, ..., n$) can be used for fault detection thresholding:

$$I^i_{\hat{y},k+1|k} = [\mu^i_{k+1} - q \times \sigma^i_{k+1}, \mu^i_{k+1} + q \times \sigma^i_{k+1}], \tag{4.17}$$

where the μ^i_{k+1} is the i_{th} component of the *a priori* output estimate, $\hat{y}_{k+1|k}$, σ^i_{k+1} is the standard deviation of $\hat{y}_{k+1|k}$ and $q = 3$ (to obtain a confidence interval at 99.7%). Thus the probability of all values lying within the thresholds is higher or equal to 99.7%.

This threshold definition can be extended to the set-membership context by using the interval expectation and the covariance matrix obtained by the iIKF. Thus $\boldsymbol{I}^i_{\hat{\boldsymbol{y}},k+1|k}$ is given, for $i = 1, ..., n$, by:

$$\boldsymbol{I}^i_{\hat{\boldsymbol{y}},k+1|k} = \left[\underline{\left(\hat{\boldsymbol{y}}^i_{k+1|k} - q \times \boldsymbol{\sigma}^i_{k+1}\right)},\ \overline{\left(\hat{\boldsymbol{y}}^i_{k+1|k} + q \times \boldsymbol{\sigma}^i_{k+1}\right)}\right], \tag{4.18}$$

where $\hat{\boldsymbol{y}}^i_{k+1|k}$ is the i_{th} component of $\hat{\boldsymbol{y}}_{k+1|k}$ and $\boldsymbol{\sigma}^i_{k+1}$ represents the standard deviation of $\hat{\boldsymbol{y}}^i_{k+1|k}$, which is approximated by the *a priori* measurement error standard deviation.

Notice that the confidence interval $\boldsymbol{I}^i_{\hat{\boldsymbol{x}},k+1|k}$ and $\boldsymbol{I}^i_{\hat{\boldsymbol{y}},k+1|k}$ are larger than any of the corresponding confidence intervals of the candidate values of the interval estimate.

It is also possible to obtain the tolerated interval of output prediction by using the Equation (4.14). We denote $\boldsymbol{I}^*_{\hat{\boldsymbol{y}},k+1|k}$ as the alternative tolerated interval:

$$\boldsymbol{I}^*_{\hat{\boldsymbol{y}},k+1|k} = \boldsymbol{C}\boldsymbol{I}_{\hat{\boldsymbol{x}},k+1|k} + \boldsymbol{D}u_{k+1}, \tag{4.19}$$

the difference of $\boldsymbol{I}^{*}_{\hat{\boldsymbol{y}},k+1|k}$ and $\boldsymbol{I}_{\hat{\boldsymbol{y}},k+1|k}$ is that the former is evaluated by using the updated state error covariance, where the measurement noise is already considered within the complete Kalman filter framework; the latter however, is calculated by using the pre-required statistical information of the noise. They can be used in different fault detection methods.

Intuitively the Equation (4.17) and (4.19) provide extended domain to any of the candidate output predictions. We can use them as thresholds for fault detection as they are adapted to parameter uncertainties, when the parameter bounds change. On the other hand, the detection thresholds result from the propagation of the parameter uncertainties within the model. That is why we call it *Adaptive threshold* based on iIKF.

4.3.2 Principle for fault detection based on adaptive threshold

We have shown that the state estimated from the iIKF approximates well the real state, if the system is not faulty and the model is consistent with the process. The system output estimate based on the state observer is also consistent with the output. Since the iIKF approach accounts for parameter uncertainty as well as Gaussian noise, the output estimate is the set of all possible outputs driven by the uncertain stochastic model. This is a very important foundation to avoid false alarms.

The output prediction $\hat{\boldsymbol{y}}_{k+1|k}$ is obtained by the *a priori* state estimate which is not corrected by the filter based on the output measurement. If the prediction is incompatible with the tolerated domain deduced from measured output and known statistical information, then the corresponding model, representing the healthy mode does not reflect the current situation of the system.

Fault detection by checking the interval intersection Fault detection can be achieved at a time $k+1$ by checking for consistency the confidence interval (at 99,7%) of each component of the measured output $y_{m,k+1}$ against the confidence interval associated to each component of the a priori estimate output $\hat{\boldsymbol{y}}_{k+1|k}$. These are consistent when their intersection is not empty. Hence a fault is detected when at least one of the component intersections is empty. Thus, we consider a binary fault indicator variable indexed by the time instant τ_{k+1}:

$$\tau_{k+1} = \begin{cases} 1 \text{ if there exists at least an index } i \text{ such that } \boldsymbol{I}^{i}_{y,m,k+1} \cap \boldsymbol{I}^{i}_{\hat{\boldsymbol{y}},k+1|k} = \emptyset, \\ 0 \text{ otherwise.} \end{cases} \tag{4.20}$$

where $\boldsymbol{I}^{i}_{y,m,k+1}$ represents the i_{th} component of the confidence interval.

Fault detection by checking the inclusion of zero Equivalently, one may test whether the interval residual includes zero or not ([Adrot 2002a]). The tolerated interval $\boldsymbol{I}^{*}_{\hat{\boldsymbol{y}},k+1|k}$ is

used in this approach and compared directly with measured output:

$$\boldsymbol{r}_{k+1} = \boldsymbol{I}^{*}_{\hat{\boldsymbol{y}},k+1|k} - y_{m,k+1}, \tag{4.21}$$

Our approach is to test if the interval residual includes zero or not, a method that can be found in [Adrot 2002a],[Marx 2010],[Makkuni 1990].

We denote $\boldsymbol{r}^{i}_{k+1}$ the i_{th} component of the vector $\boldsymbol{r}_k$. The τ_{k+1} defined previously becomes now:

$$\tau_{k+1} = \begin{cases} 1 \text{ if there exists at least an index } i \text{ such that } 0 \notin \boldsymbol{r}^{i}_{k+1} \\ 0 \text{ otherwise.} \end{cases} \tag{4.22}$$

We used Algorithm 6 (provided below) inspired by [Marx 2010].

Strategy in a fault situation The output estimate is reliable if and only if there is no faulty sensor. Indeed the occurrence of a fault invalidates the healthy model. We consider that the iIKF estimate is no longer a valid estimate for the normal system state when a fault occurs, or at least it can not be used to verify if the output sensors are healthy.

Thus as soon as the fault is detected, the innovation step in the iIKF is halted until the system is restored healthy. A similar approach can be found in [Benazera 2007, Trave-Massuyes 2001], known as the *Semi-Closed Loop* (SCL) strategy. This strategy aims at avoiding to corrupt the prediction with faulty measured data, which would then lead the prediction to "follow the fault". Our approach follows the same idea but the strategy is slightly different.

In case of faulty output, we introduce a "pure" *a priori* state estimate $\hat{\boldsymbol{x}}^{*}_{k|k-1}$ which is always evaluated based on the previous "pure" *a priori* state estimate $\hat{\boldsymbol{x}}^{*}_{k|k-1}$. The tolerated interval $\boldsymbol{I}_{\hat{y},k|k-1}$ in (4.18) is expressed as:

$$\begin{cases} \boldsymbol{I}^{i}_{\hat{\boldsymbol{y}},k|k-1} = \left[\underline{\left(\hat{\boldsymbol{y}}^{i}_{k|k-1} - q \times \boldsymbol{\sigma}^{i}_{k}\right)},\ \overline{\left(\hat{\boldsymbol{y}}^{i}_{k|k-1} + q \times \boldsymbol{\sigma}^{i}_{k}\right)}\right], \forall i \ if \ \tau^{i}_{k} = 0, \\ \boldsymbol{I}^{i}_{\hat{\boldsymbol{y}},k|k-1} = \left[\underline{\left(\hat{\boldsymbol{y}}^{i,*}_{k|k-1} - q \times \boldsymbol{\sigma}^{i}_{k}\right)},\ \overline{\left(\hat{\boldsymbol{y}}^{i,*}_{k|k-1} + q \times \boldsymbol{\sigma}^{i}_{k}\right)}\right], \forall i \ if \ \tau^{i}_{k} = 1, \\ \hat{\boldsymbol{y}}^{*}_{k|k-1} = \boldsymbol{C}\hat{\boldsymbol{x}}^{*}_{k|k-1} + \boldsymbol{D}u_k, \\ \hat{\boldsymbol{x}}^{*}_{k|k-1} = \boldsymbol{A}\hat{\boldsymbol{x}}^{*}_{k-1|k-2} + \boldsymbol{B}u_{k-1}, k \in [k_f, k_r], \end{cases} \tag{4.23}$$

where k_f is the time when the fault occurs, and k_r is the time when the system restores to healthy, $\hat{\boldsymbol{y}}^{i,*}_{k|k-1}$ is the i_{th} component of $\hat{\boldsymbol{y}}^{*}_{k|k-1}$.

The same strategy can be used on the tolerated interval $\boldsymbol{I}^*_{\hat{y},k|k-1}$ in (4.19):

$$\begin{cases} \boldsymbol{I}^i_{\hat{\boldsymbol{y}},k|k-1} = \boldsymbol{C}^{(i,:)}\boldsymbol{I}_{\hat{\boldsymbol{x}},k|k-1} + \boldsymbol{D}^{(i,:)}u_k, \forall i \; if \; \tau^i_k = 0, \\ \boldsymbol{I}^i_{\hat{\boldsymbol{y}},k|k-1} = \boldsymbol{C}^{(i,:)}\boldsymbol{I}^*_{\hat{\boldsymbol{x}},k|k-1} + \boldsymbol{D}^{(i,:)}u_k, \forall i \; if \; \tau^i_k = 1, \\ \boldsymbol{I}^{i,*}_{\hat{\boldsymbol{x}},k|k-1} = \left[\underline{\left(\hat{\boldsymbol{x}}^{i,*}_{k|k-1} - q \times \sqrt{\boldsymbol{P}^{(i,i)}_{k|k-1}}\right)}, \overline{\left(\hat{\boldsymbol{x}}^{i,*}_{k|k-1} + q \times \sqrt{\boldsymbol{P}^{(i,i)}_{k|k-1}}\right)}\right], \\ \hat{\boldsymbol{x}}^*_{k|k-1} = \boldsymbol{A}\hat{\boldsymbol{x}}^*_{k-1|k-2} + \boldsymbol{B}u_{k-1}, \\ \boldsymbol{P}^{(i,i)}_{k|k-1} = \boldsymbol{P}^{(i,i)}_{k_f}, k \in [k_f, k_r], \end{cases} \tag{4.24}$$

where $\hat{\boldsymbol{x}}^{i,*}_{k|k-1}$ is the i_{th} component of $\hat{\boldsymbol{x}}^*_{k|k-1}$, $\boldsymbol{C}^{(i,:)}$ and $\boldsymbol{D}^{(i,:)}$ are the i_{th} rows of matrices $\boldsymbol{C}$ and $\boldsymbol{D}$, respectively.

The situation of the system is defined by the following rules:

$$\begin{cases} \forall i \; such \; that \; \tau^i_{k_f-1} = 0 \; AND \; \exists i \; such \; that \; \tau^i_{k_f} = 1, \Rightarrow \; there \; is \; a \; fault \; on \; sensor \; i, \\ \exists i \; such \; that \; \tau^i_{k_r-1} = 1 \; AND \; \forall i \; such \; that \; \tau^i_{k_r} = 0. \Rightarrow \; the \; system \; restores \; to \; healthy. \end{cases}$$

In this way, the output prediction is evaluated based on the previous system status. If the sensors are detected healthy at the previous iteration, the current output is predicted from *a posteriori* state estimate.

We also set a counter Θ to ignore the temporary overstep due to unbounded noise. If the value goes back to normal before the number of time steps indicated by the counter (in the simulation we set it to 1), no alarm is triggered. This mechanism can help distinguishing excessive noise and a persistent additive faulty value.

We should notice that if the additive value from a faulty sensor is too small, it can not be distinguished from bounded tolerance defined with model, thus to be detectable a fault should be superior to the width of residual interval, which should at least fulfil:

$$f_a > w(\boldsymbol{r}_{k+1}).$$

In the next section, let us use an example to illustrate the scenario of fault detection based on the iIKF state observer.

4.3.3 Case study 1: Academic case

Let us consider the system used in Chapter 3.

$$\begin{cases} x_{k+1} = \boldsymbol{A}x_k + w_k, \\ y_k = \boldsymbol{C}x_k + v_k, \; k = 0, 1, 2, ... \end{cases}$$

Algorithm 6 Fault detection based on iIKF

Input: Variance vector $\boldsymbol{q}$ and $\boldsymbol{r}$ for noise sequence w and v respectively;
system interval parameters $\boldsymbol{A}$,$\boldsymbol{B}$,$\boldsymbol{C}$ and $\boldsymbol{D}$;
actuator command u and output measurement y

1: Define the initial state and model at instant k=0; Initialize the observer and set $k = 1$
2: Compute the *a priori* state estimate $\hat{\boldsymbol{x}}_{k|k-1}$ from *a posteriori* state estimate $\hat{\boldsymbol{x}}_{k-1|k-1}$ from time $k-1$
3: Compute the *a priori* output $\hat{\boldsymbol{y}}_{k|k-1}$ based on $\hat{\boldsymbol{x}}_{k|k-1}$
4: Collect the data u_k and y_k
5: Compute the residual $\boldsymbol{r}_k$ or the intersection $I^i_{y_k} \cap I^i_{[\hat{y}_{k|k-1}]}$
6: Compute the fault indicator σ_k according to the residual or the intersection result
7: **if** $\forall i,\ \sigma^i = 0$ **then**
8: The system is healthy, a priori state estimate $\hat{\boldsymbol{x}}_{k|k-1}$ is reliable
9: If the innovation step is halted, resume it
10: Compute the a posteriori state estimate $\hat{\boldsymbol{x}}_{k|k}$ by using IKF from a priori state estimate $\hat{\boldsymbol{x}}_{k|k-1}$
11: **end if**
12: **if** $\exists i,\ \sigma^i = 1$ **then**
13: The measurement sensor on i_{th} outputs is faulty; $\forall i$ such that $\sigma^i = 1$, the measurement y^i_k is not reliable.
14: If the innovation step is not halted, pause it; the measurement y^i is not reliable as long as the system is faulty.
15: the a posteriori state estimate equals a priori state estimate
16: **end if**
17: Increase k=k+1 and go to Step 2

where w_k et v_k are noises modelled by known Gaussian distribution, $\boldsymbol{A} = A + \triangle A, \boldsymbol{C} = C + \triangle C$,

$$A = \begin{bmatrix} 0.4 & 0.1 \\ -0.1 & 0.2 \end{bmatrix}, C = \begin{bmatrix} 0 & 1 \end{bmatrix}, Q = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}, \ R = 1.$$
$$\triangle A = \begin{bmatrix} [-0.1, 0.1] & [-0.15, 0.15] \\ 0 & [-0.25, 0.25] \end{bmatrix}, \triangle C = \begin{bmatrix} 0 & [-0.1, 0.1] \end{bmatrix},$$

The initial conditions are:

$$E\{x_0\} = \begin{bmatrix} x_{01} \\ x_{02} \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix},$$
$$Cov\{x_0\} = \begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix} = \begin{bmatrix} 0.5 & 0.0 \\ 0.0 & 0.5 \end{bmatrix}.$$

The covariance matrices are considered static without uncertainty and time invariant. The arising problem is to determine the additive fault on the measured outputs as soon as possible, although the system has uncertain parameters.

The simulation is done on the horizon [0,100].

4.3.3.1 Situation without fault

Firstly let us examine the output estimation under the normal operation condition, i.e. there is no fault. In this circumstance, the iIKF provides continuously the state estimate.

We can see in the figures 4.2, that the output estimates follow well the real measured outputs. The output prediction uses standard derivation to define the tolerated thresholds.

4.3.3.2 Situation with fault

A sensor fault affecting the system is introduced at time $k = 50$. This fault is persistent until time $k = 80$. The fault value is set to approximatively 4 standard deviations. To calculate the interval vector, we choose $q = 3$.

The residual evolution is represented on Figures 4.3 along with the output prediction and the real measured output. The obtained results are discriminating with respect to the system status. By examining the evolution of τ_k, we can conclude to the occurrence of a fault at time $k = 50$ and this fault is persistent until $k = 80$.

In lower plot of the Figure 4.3, we can see that the estimated output envelops the faulty measured output. This is due to the correction in the innovation step of iIKF. The *a posteriori* state estimate is indeed compensated according to the measured output providing an erroneous state estimate, as it is shown in Figure 4.4.

We should point out that in our scenario, no calibration takes place. But in more complex systems, it is more likely to have singular interval matrices triggering the calibration.

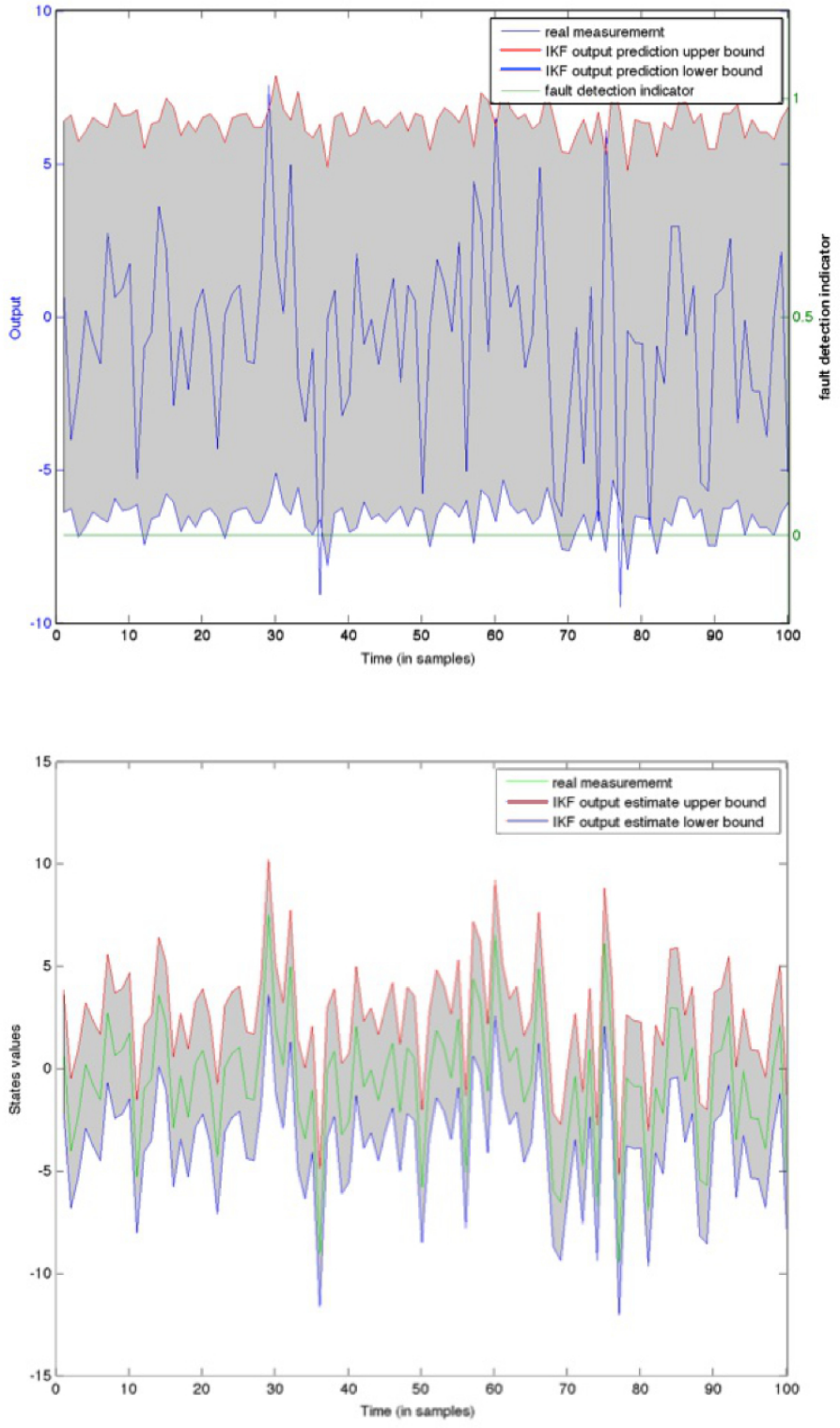

Figure 4.2: Fault detection by using the output prediction, no fault, case 1 (up) / Output based on *a posteriori* state estimate, no fault, case 1 (down).

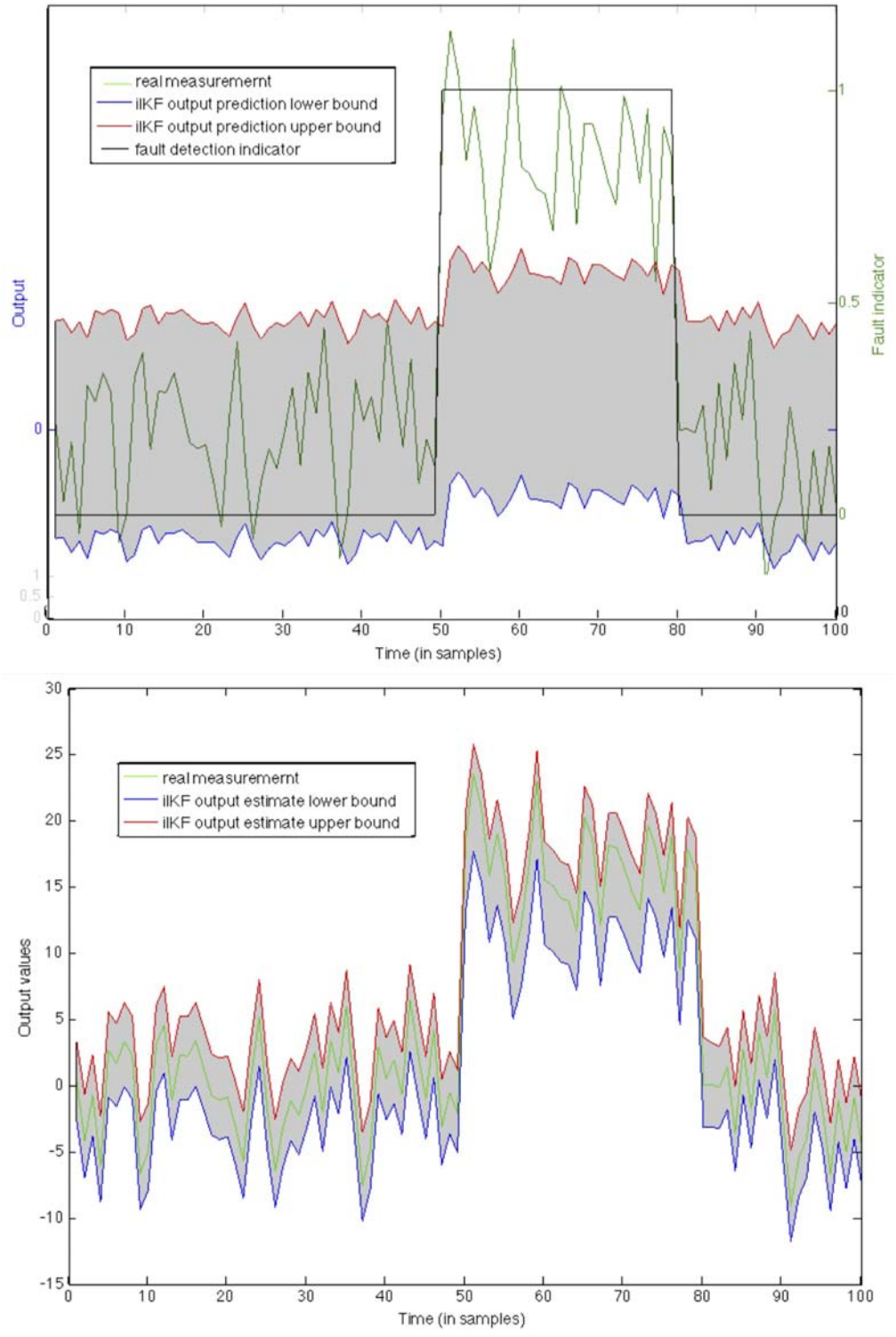

Figure 4.3: Fault detection by using the output prediction, case 1 (up) / Output *a posteriori* estimate in the faulty situation, case 1 (down).

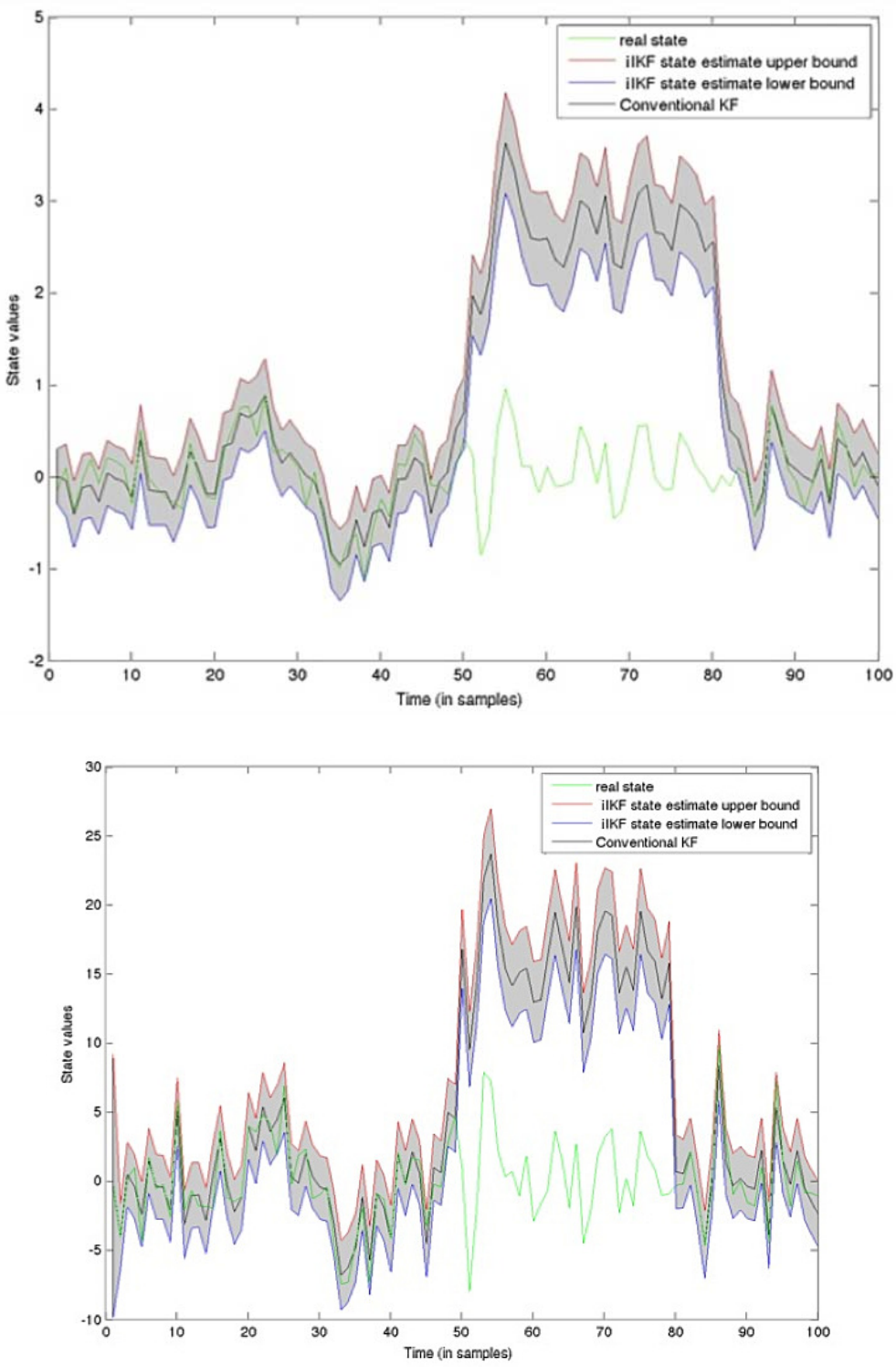

Figure 4.4: A posteriori estimate of system state 1(up) / A posteriori estimate of system state 2(down).

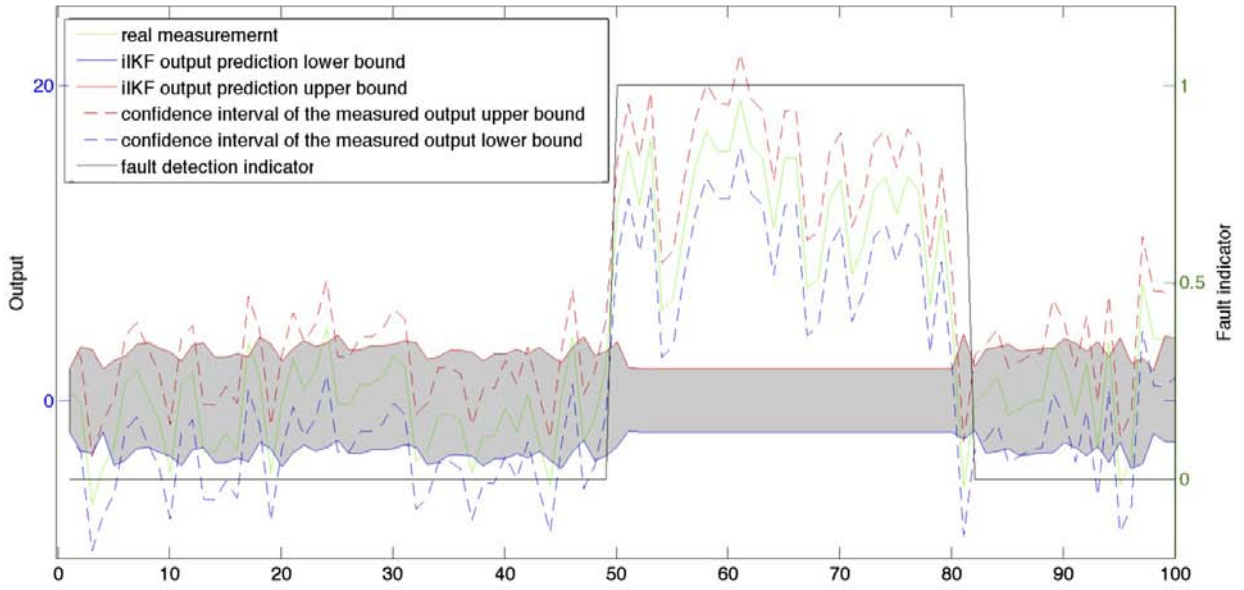

Figure 4.5: Fault detection by using the output prediction, the intersection test approach

Another approach of fault detection is by testing the intersection of the confidence interval of the measured and predicted outputs. The result of such approach is given in Figure 4.5.

The evolution of τ_k shows that the system detects the occurrence of the sensor fault at time $k = 50$, where the intersection of two confidence intervals is empty, and empty intersection persists in two consecutive steps which gives the counter $\Theta > 1$. A fault on sensor is indicated.

The measurements $y_k, k > 50$, are no longer reliable, the output prediction $\hat{\boldsymbol{y}}_{k|k-1}$ is calculated from the "pure" *a priori* system state $\hat{\boldsymbol{x}}^*$ from Equation (4.23). $\hat{\boldsymbol{x}}^*$ is not corrected according to the measurements , so it is "smoother" than the state estimate from the filter as shown in Figure 4.5. The output prediction in the faulty situation does not correspond to the real evolution of the system with the presence of noise, but it still shows the "trend" of the system evolution. If the sensor noise is centered gaussian, this output prediction in faulty case can be used to verify if the system restores to healthy.

At time $k = 80$, the intersection of two confidence intervals is no longer empty and non-empty intersection persists in two consecutive steps. A health restoration is required for the sensor.

4.3.4 Case study 2: simplified satellite altitude control system

Satellites are typical examples for autonomously operating robotic systems. The Attitude Control System (ACS) for the three-axes stabilization of a satellite is critical for its autonomous operation [Olive 2010]. We therefore use a (simplified) model of an ACS that

includes nominal parameters and uncertainties to test the fault detection capability of our iIKF approach. This case study is taken and adapted from [Bayoudh 2011]. The original case study is an Attitude and Orbit Control system that aims to stabilize the satellite attitude in the presence of disturbances by pointing the axes of the spacecraft in the direction required for its mission. A detailed description of the modelling process can be found in [Rienmüller 2013]. The system model that we used is the propulsion system. The state space representation is the following:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) + w, \\ y(t) = Cx(t) + D + v. \end{cases}$$

where w et v are noises modelled by known Gaussian distribution. Notice that the original system is continuous, and has 6 states, 3 inputs and 3 outputs. The inputs are for the purpose of controlling the system. D is a static gain. The vector of the continuously valued state variables consists of the Cardan angles and their first derivative with respect to time.

In this study, the command free behaviour of the system, given an initial state, is analysed. Among the 6 states, 3 states are controllable and the system can be simplified and reduced to 3 states, 3 outputs and command free. According to the general knowledge about aerospace applications, we choose a sample time of 0.25 s, apply the discretization [2] on the original system and obtained following system:

The original system matrices are given by:

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & \omega_0\left(\frac{I_Y - I_X}{I_Z} - 1\right) \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \omega_0\left(\frac{I_Y - I_X}{I_Z} + 1\right) & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{-I_W}{I_Z} & 0 & 0 \\ 0 & \frac{-I_W}{I_Y} & 0 \\ 0 & 0 & \frac{-I_W}{I_X} \end{bmatrix},$$

$$C = \begin{bmatrix} -\omega_0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & \omega_0 & 1 & 0 & 0 \end{bmatrix}, D = \begin{bmatrix} 0 \\ -\omega_0 \\ 0 \end{bmatrix}$$

where approximation values used for the simulation are $I_X = 600; I_Y = 700; I_Z = 600; I_W = 0.1322; \omega_0 = 2e^{-2}$, where I_X, I_Y, I_Z are the satellite inertia of each axis from the target attitude frame (Figure 4.6), I_W is the wheel inertia, ω_0 is the rotational velocity of the satellite about Y-axis. For every channel of states and outputs, the noises follow normal distributions: $w \sim N(0, 0.01)$ and $v \sim N(0, 0.01)$, which are simulated according to the general knowledge about aerospace applications.

2. The conversion from continuous-time system to discrete time is done by using the "c2d" command of Matlab with the default "zero-order hold on the inputs" method.

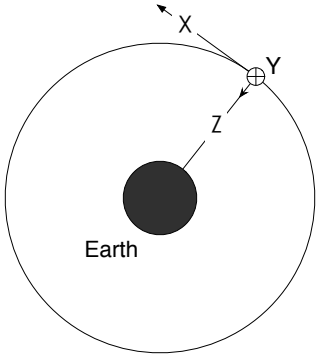

Figure 4.6: Target attitude frame $C = (X, Y, Z)$ that depends on the satellite mission.

After discretization with sample time T=0.25s, we obtain:

$$A_d = \left[\begin{array}{ccc|ccc} 1 & 0 & 0 & 0.25 & 0 & -0.00052 \\ 0 & 1 & 0 & 0 & 0.25 & 0 \\ 0 & 0 & 1 & 0.00052 & 0 & 0.25 \\ - & - & - & + & & \\ 0 & 0 & 0 & 1 & 0 & -0.0041 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0.0041 & 0 & 1 \end{array}\right], C_d = \begin{bmatrix} -0.02 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0.02 & 1 & 0 & 0 \end{bmatrix}, D_d = \emptyset$$

B_d is not given because we analyse the system behaviour when it is command free. In addition, as the 4th to 6th states are independent from the first three states (cf. the block diagonal structure of A_d), we decided to build a reduced satellite altitude control system based on the 4th to 6th states as follows:

$$\begin{cases} x_{k+1} = A_d^* x_k + w_k, \\ y_k = C_d^* x_k + v_k, k = 0, 1, 2, ... \end{cases}$$

where

$$A_d^* = \begin{bmatrix} 1 & 0 & -0.0041 \\ 0 & 1 & 0 \\ 0.0041 & 0 & 1 \end{bmatrix}, C_d^* = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

with

$$Q = R = \begin{bmatrix} 0.001 & 0 & 0 \\ 0 & 0.001 & 0 \\ 0 & 0 & 0.001 \end{bmatrix}.$$

and the initial conditions are:

$$E\{x_0\} = \begin{bmatrix} x_{01} \\ x_{02} \\ x_{03} \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, Cov\{x_0\} = \begin{bmatrix} 0.1 & 0.0 & 0.0 \\ 0.0 & 0.1 & 0.0 \\ 0.0 & 0.0 & 0.1 \end{bmatrix}.$$

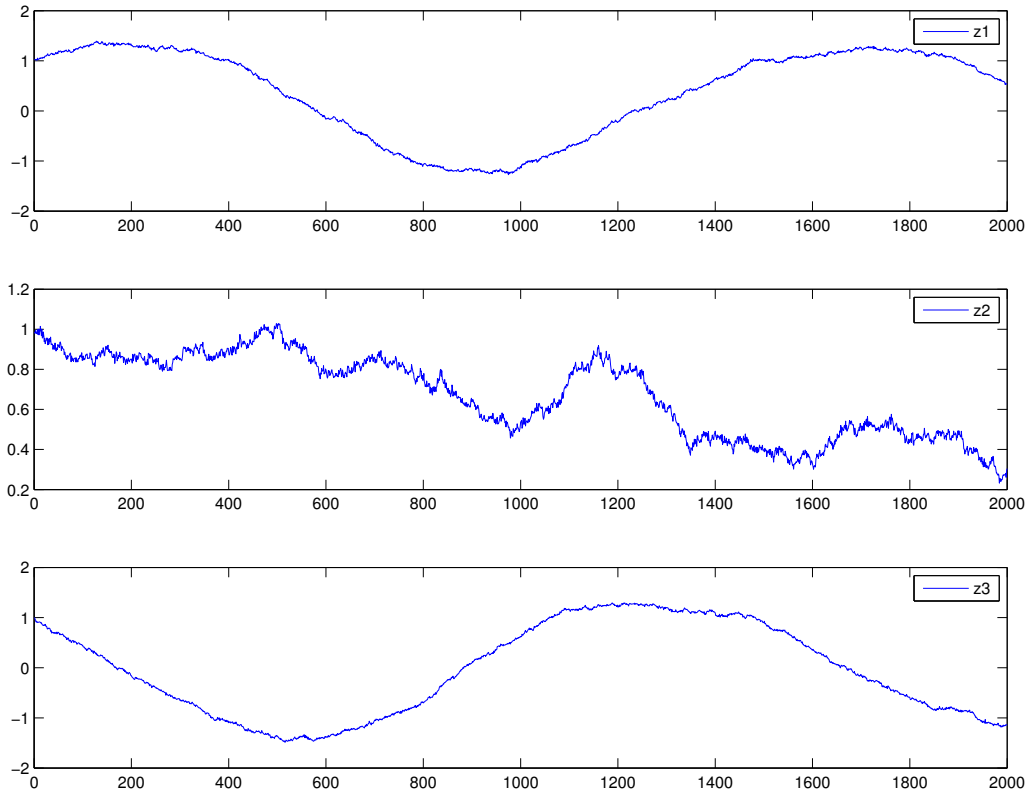

Figure 4.7: Measurement simulation of the satellite altitude control system example

In the simulation, the response of the system without parameter uncertainty is shown in Figure 4.7 on the horizon [0,2000]

The system 1st and 3rd outputs in the command free mode are sinus signals, while 2nd signal is a total free response with only the noise affection.

In order to test our fault detection algorithm, parameter uncertainty is introduced to this system according to the general knowledge about aerospace applications, by setting $\boldsymbol{A} = A_d^* + \triangle A$ with:

$$\triangle A = \begin{bmatrix} 0 & 0 & [-0.0001, 0.0001] \\ 0 & 0 & 0 \\ [-0.0001, 0.0001] & 0 & 0 \end{bmatrix}.$$

The same scenario implemented in the first case study is considered; a sensor fault affects the system in all the measured output channels, beginning at step k=50, and the system is restored at step k=80. The fault value is set to about 4 times the standard deviation.

For residual interval, we also choose $q = 3$ as the multiplicative factor of the standard deviation, which covers 97% of the possible noises.

The residual evolutions are represented in the upper plot of Figure 4.8 along with the output prediction and the real measured outputs. The lower plot of Figure 4.8 shows the evolution of system state estimates. The obtained results are discriminating with respect to the system status. The examination of the fault indicator τ_k enables to conclude to the occurrence of the fault in the time interval [50,80], the fault is detected and the alarm persists during the faulty situation.

Another approach of fault detection is testing the intersection of the confidence interval of the measured output and predicted output. The results of such approach applied on this case study are given in Figure 4.9.

The upper plot concerns a single sensor fault on output 1, from time $k = 50$ to $k = 80$. The evolution of τ_k shows that the system detects the occurrence of the sensor fault on output 1 at time $k = 50$, for which the intersection of the two confidence intervals is empty, and empty intersection persists in two consecutive steps. A fault on the output 1 sensor is raised.

The measurements $\boldsymbol{y}^1_{m,k}, k > 50$ are no longer reliable, but the other output sensors are still sane. Only the output prediction $\hat{\boldsymbol{y}}^1_{k|k-1}$ is calculated from the "pure" *a priori* system state $\hat{\boldsymbol{x}}^*$ from Equation (4.23). $\hat{\boldsymbol{x}}^*$ is not corrected according to the measurements and knowledges about the noise, so it is "smoother" than the state estimate from filter. The other output predictions are still evaluated normally. The output prediction in faulty situation does not correspond to the real evolution of the system with the presence of noise, but it still shows the "trend" of the system evolution. When the sensor noise is centered gaussian, the output prediction in faulty case can be used to verify if the system restores to healthy.

At time $k = 80$, the intersection of two confidence intervals is no longer empty during $\Theta = 2$. A health restoration on output 1 sensor is raised.

The lower plot concerns a multi sensor fault on 1st output from time $k = 50$ to $k = 80$, and on 3rd output from time $k = 30$ to $k = 60$. The evolution of τ_k shows that the system detects the occurrence of sensor fault on output 3 at time $k = 30$, and the occurrence of the sensor fault on 1st output at time $k = 50$, separately. Indeed the intersections of two confidence intervals are empty, and empty intersections persist in two consecutive steps. Different faults on sensors of output 1 and 3 are detected correctly.

The measurements $\boldsymbol{y}^1_k, k > 50$ and $\boldsymbol{y}^3_k, k > 30$ are no longer reliable, but the output sensor 2 is still sane. The output prediction $\hat{\boldsymbol{y}}^1_{k|k-1}$ and $\hat{\boldsymbol{y}}^3_{k|k-1}$ are calculated from the "pure" *a priori* system state $\hat{\boldsymbol{x}}^*$ (see Equation (4.23)). The prediction of the output 2 is still evaluated normally. The output predictions in the faulty situation do not correspond to the real evolution of the system with the presence of noise, but they still show the "trend" of the system evolution. When the sensor noise is centered gaussian, the output prediction in the faulty case can be used to verify if the system restores to healthy.

At time $k = 60$ and $k = 80$, the intersections of two confidence intervals are no longer

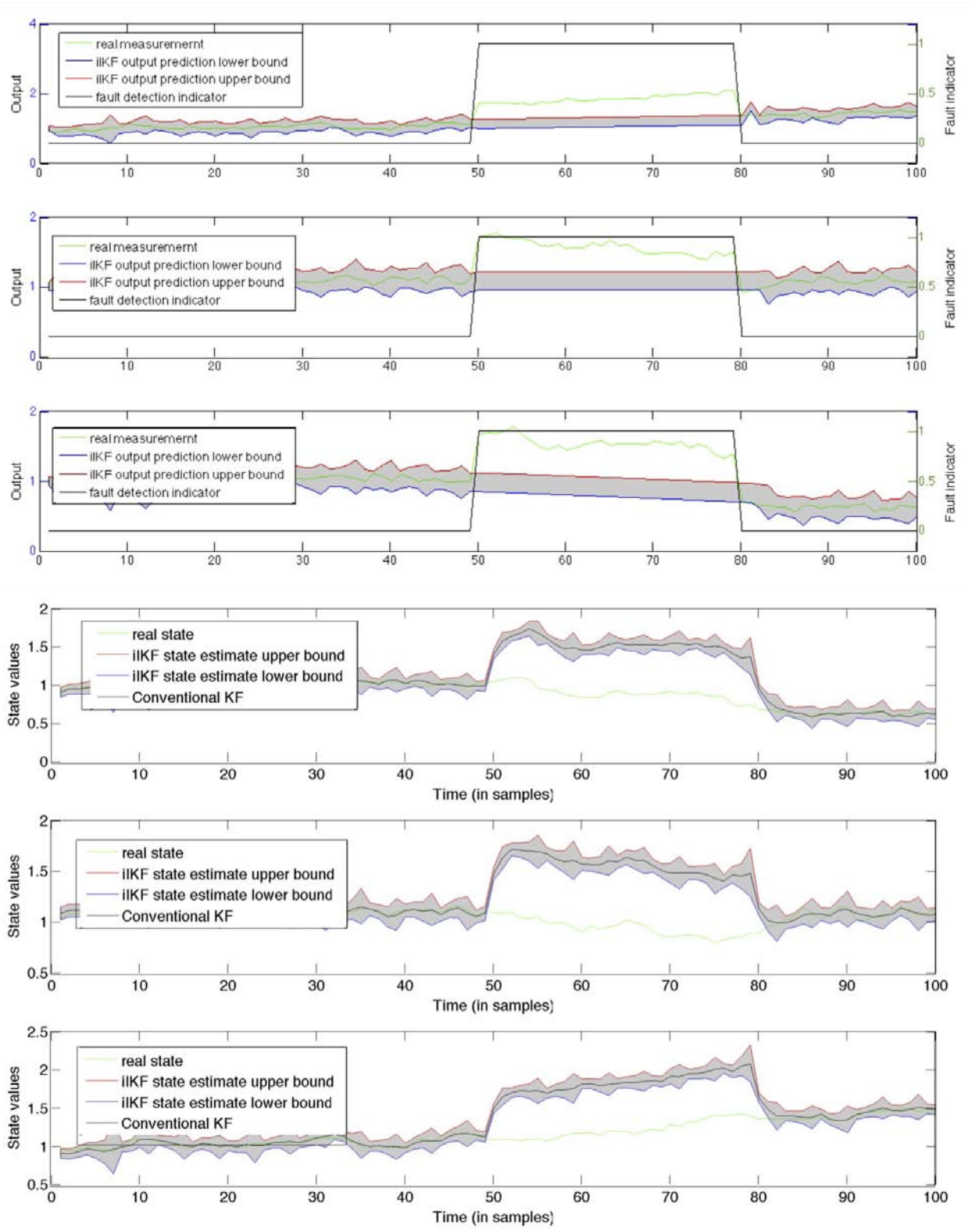

Figure 4.8: Fault detection by using the output prediction (up) / A posteriori state estimate(down).

empty during $\Theta = 2$. The health restoration on sensors 1 and 3 are both indicated.

The simulations show that our approaches of fault detection are quite sensitive to the occurrence of faults and also the disappearance of faults; They can detect multiple faults on different sensors even when they are present in the system on overlapping time windows.

4.4 Conclusion

In this chapter, the basic principle of fault detection using Kalman filters for systems without parameter uncertainty is firstly reviewed. The approach that inverses the measured output equation to define the admissible domain is then presented for cases in which parameter uncertainty and output noises are both considered as known bounded values.

Our work consists in proposing a solution for system with the bounded parameter uncertainty and centered gaussian noise, i.e. stochastic uncertain systems. To do so, the basic residual generation method based on the conventional Kalman filter is kept, i.e. we use the output prediction as the reference of healthy system, and use the standard deviation to build the fault detection threshold. Different from the fault detection based on the conventional Kalman filter, the output prediction from iIKF contains all the possible values from parameter uncertainty and the threshold is actually adapted to the size of the parameter intervals. When the parameter intervals are large, the threshold is loose, which can avoid false alarms when the parameters are not precisely known; when the parameter intervals are narrow, the threshold does not change much from the one of the conventional Kalman filter, which prevents to miss alarms. That is why we call it adaptive threshold based fault detection.

The iIKF is used to propose a fault detection algorithm which makes use of a Semi-Closed Loop strategy as proposed by [Benazera 2007, Trave-Massuyes 2001]. Through an example, the advantages of the iIKF with respect to previous versions are exhibited and the efficiency of the iIKF based Semi-Closed Loop fault detection algorithm is clearly demonstrated. The two considered cases have shown that the proposed method is efficient to detect the occurrence of an additive measurement fault.

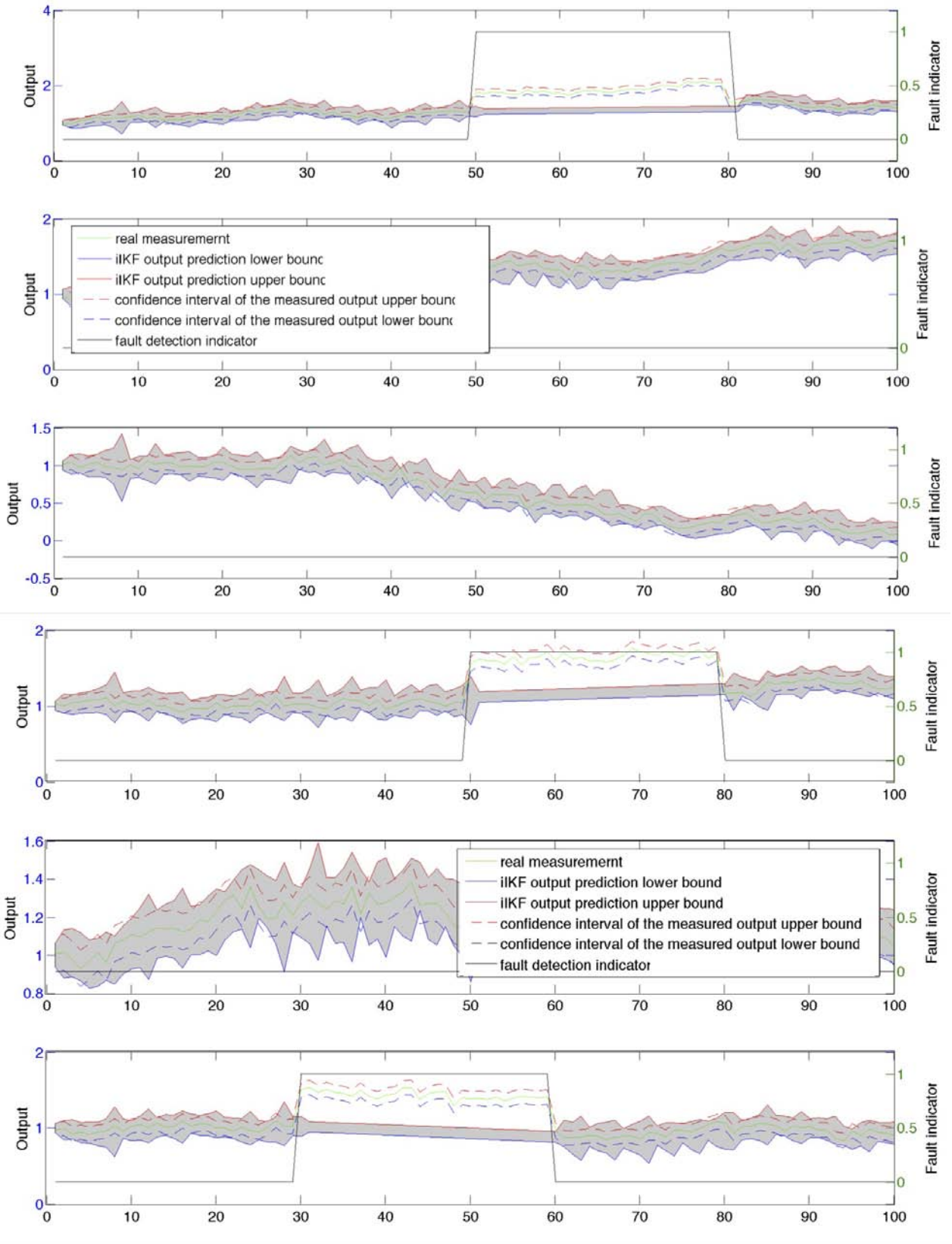

Figure 4.9: Fault detection by using the intersection approach: single fault in one output (up) / Fault detection by using the intersection approach: multiple fault in different outputs at different times (down).

Conclusion and perspectives

In this thesis, a new approach to estimation problems under the presence of bounded uncertain parameters and statistical noise has been presented. The objective is to use the uncertainty model which appears as the most appropriate for every kind of uncertainty. This leads to the need to consider uncertain stochastic systems and to study how the two types of uncertainty combine.

In the first chapter, the concept of fault as known in the diagnosis field is recalled and the different kinds of uncertainties are overviewed. Several state/parameter estimation methods are presented. Considering the problem of combining statistic and bounded uncertainties, statistical noise is modeled as the centered gaussian variable and the unknown but bounded parameters are approximated by intervals. This results in an estimation problem that demands the development of mixed filters and a set-theoretic strategy.

In chapter 2, the basics of interval analysis are introduced. The attention is drawn on set inversion problems and constraint satisfaction problems. The former is the foundation of a method for solving interval equations, and the latter can significantly improve the speed of interval based arithmetic and algorithms. The interval matrix inversion problem is also discussed. An important contribution of this work consists in proposing an interval matrix inversion method which couples the algorithm SIVIA with the construction of a list of constraint propagation problems.

The uncertain stochastic state estimation problem is mainly discussed in chapter 3. The system model is formalized as an uncertain stochastic system. Starting with the interval Kalman filtering algorithm proposed in [Chen 1997] and that we name the IKF, an improved interval Kalman filtering algorithm (iIKF) is proposed. This algorithm is based on interval conditional expectation for interval linear systems, which is demonstrated in the beginning of the chapter. The iIKF has the same structure as the conventional Kalman filter while achieving guaranteed statistical optimality. The recursive computational scheme is developed in the set-membership context. Our improvements achieve guaranteed interval inversion whereas the original version IKF [Chen 1997] uses an instance (the upper bound) of the interval matrix to avoid the possible singularity problems. This point of view leads to a sub-optimal solution that does not preserve guaranteed results, some solutions being lost. On the contrary, in the presence of unknown-but-bounded parameters and measurement statistical errors, our estimation approach in the form of the iIKF provides guaranteed estimates, while maintaining a computational burden comparable to that of classic statistical approaches. Several constraint based techniques have also been implemented to limit the overestimation effect due to interval propagation within the interval Kalman filter recursive structure:

– constraints on the interval covariance matrix have been added to make sure that the covariance matrix respects the fact that its diagonal elements are semi-positive;

- the gain of the filter is obtained by a calculus based on the set inversion algorithm SIVIA complemented by constraint propagation;
- a constraint propagation algorithm has been implemented based on the forward-backward method, and also an interval intersection rule has been used to find a more compact result for the product of n interval matrices;
- based on the system state estimate calculated from the initial state, another constraint on the width of the iIKF state estimate has been added. A calibration has been implemented to reset the iteration for limiting divergence once the width of the iIKF estimate exceeds a given threshold.

The results have shown that the iIKF outputs bounded estimates that enclose all the solutions consistent with bounded errors and achieves good overestimation control.

In chapter 4, The iIKF is used to propose a fault detection algorithm which makes use of a Semi-Closed Loop strategy which does not correct the state estimate with the measure as soon as a fault is detected. Two methods for generating fault indicators are proposed: they use the a priori state estimate and a threshold based on the a posteriori and a priori covariance matrix, respectively, and check the consistency against the measured output. Through different examples, the advantages of the iIKF with respect to previous versions are exhibited and the efficiency of the iIKF based Semi-Closed Loop fault detection algorithm is clearly demonstrated.

Further improvements should target to improve the algorithm efficiency and speed. The constraint propagation is still an important direction in order to achieve higher performance. Additional constraints can probably be added, so that constraint propagation techniques further reduce the estimate interval width and to improve the algorithm efficiency.

Another issue for future work is to theoretically unify the combination of interval variables and statistical variables. One of the difficulties in uncertain stochastic systems is that there exist two different types of variables at the same time. Our approach is to superpose the stochastic support (centered gaussian distributed values) on known intervals, which gives us a new gaussian variable that has interval expectation and variance. But we also proved that a gaussian variable with interval expectation and variance can be approximated by another gaussian variable with the middle value of the interval expectation and variance. That rises a different point of view about bounded variables. It can be interpreted as a uniformly distributed variable on a bounded support. With this approach, the uncertainties may be entirely unified under the statistical framework.

Finally, this work has concentrated on linear discrete systems. So, there is space for extensions to non linear systems, as it is shown in [Abdallah 2008]. In this case, a interesting direction for state estimation could be to consider uncertain particle filtering for which the prediction phase could use bounded uncertainty models. As a consequence, the resampling phase should account for the uncertain predictions and their overlaps.

APPENDIX A

Estimation based on the initial state and historical command and noises

The recursive estimator:

$$x_{k+1} = (A + \Delta A)x_k + (B + \Delta B)u_k + w_k. \tag{A.1}$$

can be rewritten based on the initial state and historical command and noises.

$$x_{k+1} = \sum_{i=1}^{k+1}(A + \Delta A)^{k+1-i}w_{i-1} + (A + \Delta A)^{k+1}x_0 + \sum_{i=1}^{k+1}(A + \Delta A)^{k+1-i}(B + \Delta B)u_{i-1}. \tag{A.2}$$

The proof uses a recursive deduction:

Proof A.1 For k=0 *from A.1*

$$x_1 = (A + \Delta A)x_0 + (B + \Delta B)u_0 + w_0. \tag{A.3}$$

k=1

$$\begin{aligned} x_2 &= (A + \Delta A)x_1 + (B + \Delta B)u_1 + w_1 \\ &= (A + \Delta A)[(A + \Delta A)x_0 + (B + \Delta B)u_0 + w_0] + (B + \Delta B)u_1 + w_1 \\ &= w_1 + (A + \Delta A)w_0 + (A + \Delta A)^2x_0 + \sum_{i=1}^{2}(A + \Delta A)2 - i(B + \Delta B)u_{i-1}. \end{aligned} \tag{A.4}$$

Suppose that:

$$x_k = \sum_{i=1}^{k}(A + \Delta A)^{k-i}w_{i-1} + (A + \Delta A)^k x_0 + \sum_{i=1}^{k}(A + \Delta A)^{k-i}(B + \Delta B)u_{i-1}.$$

then from A.1:

$$\begin{aligned} x_{k+1} =& (A + \Delta A)[\sum_{i=1}^{k}(A + \Delta A)^{k-i}w_{i-1} + (A + \Delta A)^k x_0 \\ &+ \sum_{i=1}^{k}(A + \Delta A)^{k-i}(B + \Delta B)u_{i-1}] + (B + \Delta B)u_k + w_k \\ =& \sum_{i=1}^{k+1}(A + \Delta A)^{k+1-i}w_{i-1} + (A + \Delta A)^{k+1}x_0 \\ &+ \sum_{i=1}^{k+1}(A + \Delta A)^{k+1-i}(B + \Delta B)u_{i-1}. \end{aligned} \tag{A.5}$$

and A.2 holds.

Bibliography

[Abdallah 2007] F Abdallah and A Gning. *Filtrage ensembliste multi-hypothèse sur des données intervalles: Application à la localisation*. In XXI^{e} Colloque GRETSI: Traitement du signal et des images, 2007. (Cited on page 19.)

[Abdallah 2008] Fahed Abdallah, Amadou Gning and Philippe Bonnifait. *Box particle filtering for nonlinear state estimation using interval analysis*. Automatica, vol. 44, no. 3, pages 807–815, March 2008. (Cited on page 124.)

[Abellán 2005] J Abellán. *Upper entropy of credal sets. Applications to credal classification*. International Journal of Approximate Reasoning, vol. 39, no. 2-3, pages 235–255, June 2005. (Cited on page 9.)

[Adrot 2000] O Adrot, D Maquin and J Ragot. *Diagnosis of an uncertain static system*. In IEEE Conference of Decision and Control, pages 4150–4155, 2000. (Cited on page 31.)

[Adrot 2002a] O Adrot, H Janati-Idrissi and D Maquin. *Fault detection based on interval analysis*. In 15th Triennial World Congress, Barcelona, Spain, 2002. (Cited on pages 106 and 107.)

[Adrot 2002b] O Adrot, S Ploix and J Ragot. *Caractérisation des incertitudes dans un modèle linéaire statique*. Journal Européen des Systèmes Automatisés, vol. 36, no. 6, pages 799–824, 2002. (Cited on page 10.)

[Adrot 2008] O Adrot and J M Flaus. *Fault detection based on uncertain models with bounded parameters and bounded parameter variations*. In Proceedings of the 17th IFAC World Congress, volume 17, pages 7338–7343, 2008. (Cited on page 31.)

[Alamo 2003] T Alamo, J M Bravo and E F Camacho. *Guaranteed state estimation by zonotopes*. In Proceedings of the 42nd IEEE Conference of Decision and Control, pages 5831–5836, 2003. (Cited on page 24.)

[Alessandri 2000] A Alessandri. *Design of sliding-mode observers and filters for nonlinear dynamic systems*. In Proceedings of the 39th IEEE Conference of Decision and Control, pages 2593–2598, 2000. (Cited on pages 11 and 19.)

[Alhaj-Dibo 2004] M Alhaj-Dibo, D Maquin and J Ragot. *Data validation of uncertain dynamic systems*. In 15th International Workshop on Principles of Diagnosis, DX'04, Carcassonne, France, June 2004. (Cited on page 33.)

[Alvarez 2009] D Alvarez. *A Monte Carlo-based method for the estimation of lower and upper probabilities of events using infinite random sets of indexable type*. Fuzzy Sets And Systems, vol. 160, no. 3, pages 384–401, February 2009. (Cited on page 9.)

[Andrieu 2004] C Andrieu, A Doucet and S Singh. *Particle methods for change detection, system identification, and control*. In Proceedings of the IEEE, volume 92, pages 423–438, 2004. (Cited on page 18.)

[Ashokaraj 2005] I Ashokaraj, A Tsourdos, P Silson and B White. *Sensor based robot localisation and navigation: Using interval analysis and extended Kalman filter*. In 5th Control Conference, volume 2, pages 1086–1093, 2005. (Cited on page 83.)

[Batarseh 2008] O G Batarseh and Y Wang. *Reliable simulation with input uncertainties using an interval-based approach*. In Simulation Conference, pages 344–352, 2008. (Cited on page 9.)

[Bayoudh 2011] M Bayoudh and L Trave-Massuyes. *Diagnosability analysis of hybrid systems casted in a discrete-event framework*. Discrete Event Dynamic Systems Theory and Applications, pages 1–27, February 2011. (Cited on page 115.)

[Bellman 1971] R Bellman. *On the identification of systems and some related questions*. In Conference on Applications of Numerical Analysis, pages 1–8, January 1971. (Cited on page 29.)

[Benazera 2007] E Benazera and L Trave-Massuyes. *A diagnosis driven self-reconfigurable filter*. In 18th International Workshop on Principles of Diagnosis, DX'07, pages 21–28, 2007. (Cited on pages 107 and 120.)

[Bernard 2004] O Bernard and J L Gouzé. *Closed loop observers bundle for uncertain biotechnological models*. Journal of Process Control, vol. 14, no. 7, pages 765–774, October 2004. (Cited on page 23.)

[Blanchard 2007] E Blanchard, A Sandu and C Sandu. *Parameter Estimation Method using an Extended Kalman Filter*. In Proceedings of the Joint North America, Asia-Pacific ISTVS Conference and Annual Meeting of Japanese Society for Terramechanics, pages 1–14, March 2007. (Cited on page 17.)

[Bloch 1996] I Bloch. *Incertitude, imprécision et additivité en fusion de données: point de vue historique (Uncertainty, imprecision and additivity in data fusion: Historical point of view)*. Traitement du signal, vol. 13, no. 4, pages 267–288, 1996. (Cited on page 8.)

[Bourgain 1989] J Bourgain, J Lindenstrauss and V Milman. *Approximation of zonoids by zonotopes*. Acta Mathematica, vol. 162, no. 1, pages 73–141, 1989. (Cited on page 10.)

[Bourij 1999] A Bourij and D Koenig. *An original Petri net state estimation by a reduced Luenberger observer*. In Proceedings of the 1999 American Control Conference, pages 1986–1989, 1999. (Cited on page 15.)

[Bouron 2002] P Bouron. *Méthodes ensemblistes pour le diagnostic, l'estimation d'état et la fusion de donnéees temporelles.* PhD thesis, Université de Compiègne, Compiegne, France, Juillet 2002. (Cited on page 30.)

[Bravo 2004] J M Bravo, T Alamo and E Camacho. *Bounded error identification of systems with time-varying parameters*. Atlantis, vol. 51, no. 7, pages 1144–1150, 2004. (Cited on pages 22 and 30.)

[Calafiore 2004] G Calafiore and L El Ghaoui. *Ellipsoidal bounds for uncertain linear equations and dynamical systems*. Automatica, vol. 40, pages 773–787, 2004. (Cited on page 10.)

[Calderón-Espinoza 2007a] G Calderón-Espinoza, J Armengol and J Vehi. *Dynamical diagnosis with interval analytical redundancy relations*. Journal AI Communications - Model-Based Systems, vol. 20, no. 1, pages 39–47, January 2007. (Cited on page 31.)

[Calderón-Espinoza 2007b] G Calderón-Espinoza, J Armengol, J Vehi and E R Gelso. *Dynamic diagnosis based on interval analytical redundancy relations and signs of the symptoms*. AI Communications, vol. 20, no. 1, pages 39–47, 2007. (Cited on page 31.)

[Canale 2009] M Canale, L Fagiano and M Milanese. *Set Membership approximation theory for fast implementation of Model Predictive Control laws*. Automatica, vol. 45, no. 1, pages 45–54, 2009. (Cited on page 10.)

[Chabert 2009] G Chabert, L Jaulin and X Lorca. *A Constraint on the Number of Distinct Vectors with Application to Localization*. In 15th International Conference of Principles and Practice of Constraint Programming, volume 5732, pages 196–210, 2009. (Cited on pages 39 and 55.)

[Chaves 2002] M Chaves and E D Sontag. *State-estimators for chemical reaction networks of Feinberg-Horn-Jackson zero deficiency type*. In Proceedings of European Control conference, pages 343–359, December 2002. (Cited on page 15.)

[Chen 1997] G R Chen, J R Wang and L S Shieh. *Interval kalman filtering*. IEEE Transactions on Aerospace and Electronic Systems, vol. 33, no. 1, pages 250–259, 1997. (Cited on pages iii, v, 1, 39, 83, 90, 91, 93 and 123.)

[Chisci 1998] L Chisci, A Garulli, A Vicino and G Zappa. *Block recursive parallelotopic bounding in set membership identification*. Automatica, vol. 34, no. 1, pages 15–22, 1998. (Cited on page 22.)

[Chow 1984] E Chow and A Willsky. *Analytical redundancy and the design of robust failure detection systems*. IEEE Transactions on Automatic Control, vol. 29, no. 7, pages 603–614, 1984. (Cited on page 30.)

[Chui 1987] C K Chui. Kalman filtering with real-time applications. Springer, 1987. (Cited on page 14.)

[Combastel 2003] C Combastel. *A state bounding observer based on zonotopes*. In Proceedings of European Control Conference, Cambridge, UK., January 2003. (Cited on pages 22 and 24.)

[Combastel 2004] C Combastel. *Observation de systèmes non linéaires appliquée à un modèle de bioréacteur. Une approche ensembliste basée sur les zonotopes*. Journal Européen des Systèmes Automatisés, vol. 38, no. 7-8, pages 933–957, October 2004. (Cited on page 24.)

[Combastel 2005] C Combastel. *A State Bounding Observer for Uncertain Non-linear Continuous-time Systems based on Zonotopes*. In Proceedings of the 44th IEEE Conference of Decision and Control, pages 7228–7234, 2005. (Cited on page 24.)

[Corliss 1994] G F Corliss. *Guaranteed error bounds for ordinary differential equations*. In Theory of Numerics in Ordinary and Partial Differential Equations, pages 1–75, 1994. (Cited on page 50.)

[Cozman 1997] F G Cozman. *Learning convex sets of probability from data*. Rapport technique CMU-RI-TR-97-25, Robotics Institute, Carnegie Mellon University, June 1997. (Cited on page 9.)

[Crisan 2002] D Crisan and A Doucet. *A survey of convergence results on particle filtering methods for practitioners*. IEEE Transactions on Signal Processing, vol. 50, no. 3, pages 736–746, March 2002. (Cited on page 18.)

[Csendes 1997] T Csendes and D Ratz. *Subdivision direction selection in interval methods for global optimization*. SIAM Journal on Numerical Analysis, pages 922–938, 1997. (Cited on page 53.)

[Csendes 2003] T Csendes. *Numerical experiences with a new generalized subinterval selection criterion for interval global optimization*. Reliable Computing, vol. 9, no. 2, pages 109–125, 2003. (Cited on page 53.)

[Csendes 2004] T Csendes. *Generalized subinterval selection criteria for interval global optimization*. Numerical Algorithms, vol. 37, no. 1, pages 93–100, 2004. (Cited on page 53.)

[Cuzzolin 2010] F Cuzzolin. *Credal semantics of Bayesian transformations in terms of probability intervals*. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 40, no. 2, pages 421–432, 2010. (Cited on page 9.)

[Dao 2006] M Dao. *Caractérisation d'ensembles par des méthodes intervalles. Applications en automatique*. PhD thesis, Universite d'Angers, December 2006. (Cited on page 38.)

[Desai 1981] M Desai and A Ray. *A fault detection and isolation methodology*. Decision and Control including the Symposium on Adaptive Processes, pages 1363–1369, 1981. (Cited on page 26.)

[Deville 2002] Y Deville, M Janssen and P van Hentenryck. *Consistency techniques in ordinary differential equations*. Constraints, vol. 7, no. 3-4, pages 289–315, 2002. (Cited on page 23.)

[Doucet 1998] A Doucet, S J Godsill and C Andrieu. *On sequential simulation-based methods for Bayesian filtering*. Rapport technique, Dept. of Engineering, University of Cambridge, 1998. (Cited on page 18.)

[Doucet 2001a] A Doucet and N de Freitas. Sequential Monte Carlo methods in practice. Springer, 2001. (Cited on page 18.)

[Doucet 2001b] A Doucet, N J Gordon and V Krishnamurthy. *Particle filters for state estimation of jump Markov linear systems*. IEEE Transactions on Signal Processing, vol. 49, no. 3, pages 613–624, 2001. (Cited on page 18.)

[Drakunov 2010] S V Drakunov. *Sliding-mode observers based on equivalent control method*. In Conference on Control and Fault-Tolerant Systems (SysTol), pages 2368–2369. IEEE, 2010. (Cited on page 11.)

[Durola 2008] S Durola and P Danès. *Set-membership filtering of uncertain discrete-time rational systems through Recursive Algebraic Representations and LMIs*. In 47th IEEE Conference on Decision and Control, pages 684–689, 2008. (Cited on page 22.)

[Edelmayer 2007] A Edelmayer and M Miranda. *Federated filtering for fault tolerant estimation and sensor redundancy management in coupled dynamics distributed systems*. In Mediterranean Conference on Control and Automation, pages 1–6, 2007. (Cited on page 17.)

[Escobet 2001] T Escobet and L Trave-Massuyes. *Parameter estimation methods for fault detection and isolation*. In Proceedings of the 12th International Workshop on Principles of Diagnosis, pages 1–11, 2001. (Cited on page 29.)

[Evensen 2003] Geir Evensen. *The Ensemble Kalman Filter: theoretical formulation and practical implementation*. Ocean Dynamics, vol. 53, no. 4, pages 343–367, 2003. (Cited on page 17.)

[Fairman 1977] F W Fairman. *Reduced order state estimators for discrete-time stochastic systems*. IEEE Transactions on Automatic Control, vol. 22, no. 4, pages 673–675, 1977. (Cited on page 15.)

[Faubel 2008] F Faubel, H Raja and J McDonough. *Particle filter based soft-mask estimation for missing feature reconstruction*. In Proceedings of the 11th International Workshop on Acoustic Echo and Noise Control, 2008. (Cited on page 18.)

[Ferson 2002] S Ferson and L Ginzburg. *Whereof one cannot speak: when input distributions are unknown*. Risk Analysis, 2002. (Cited on page 9.)

[Frank 1994] P Frank. *On-line fault detection in uncertain nonlinear systems using diagnostic observers: a survey*. International Journal of Systems Science, vol. 25, no. 12, pages 2129–2154, 1994. (Cited on page 32.)

[Frank 1997] P Frank and X Ding. *Survey of robust residual generation and evaluation methods in observer-based fault detection systems*. Journal of Process Control, vol. 7, no. 6, pages 403–424, January 1997. (Cited on page 32.)

[Funiak 2003] S Funiak. *Multi-modal particle filtering for hybrid systems with autonomous mode transitions*. In Workshop on Principles of Diagnosis, 2003. (Cited on page 18.)

[Gamerman 1998] D Gamerman. *Markov chain Monte Carlo for dynamic generalised linear models*. Biometrika, vol. 85, no. 1, pages 215–227, March 1998. (Cited on page 18.)

[Garcia 1997] E A Garcia and P Frank. *Deterministic nonlinear observer-based approaches to fault diagnosis: a survey*. Control Engineering Practice, vol. 5, no. 5, pages 663–670, May 1997. (Cited on page 32.)

[Gardeñes 1986] E Gardeñes, H Mielgo and A Trepat. *Modal intervals: reason and ground semantics*. Interval Mathematics, vol. 212, pages 27–35, 1986. (Cited on page 44.)

[Garulli 1999] A Garulli, A Vicino and G Zappa. *Optimal induced-norm and set membership state smoothing and filtering for linear systems with bounded disturbances*. Automatica, vol. 35, pages 767–776, 1999. (Cited on page 22.)

[Gelso 2008] E R Gelso, G Biswas, S M Castillo and J Armengol. *A comparison of two methods for fault detection: a statistical decision, and an interval-based approach*. In 19th International Workshop on Principles of Diagnosis DX, pages 261–268, Blue Montains, Australia, 2008. (Cited on pages 1 and 72.)

[Gertler 1997] J Gertler. *Fault detection and isolation using parity relations*. Control Engineering Practice, vol. 5, no. 5, pages 653–661, May 1997. (Cited on page 31.)

[Gevers 1996] M Gevers. *Identification for control*. Annual Reviews in Control, vol. 20, pages 95–106, January 1996. (Cited on page 29.)

[Goldsztejn 2005] A Goldsztejn. *A Right-Preconditioning Process for the Formal–Algebraic Approach to Inner and Outer Estimation of AE-Solution Sets*. Reliable Computing, vol. 11, no. 6, pages 443–478, 2005. (Cited on page 39.)

[Goldsztejn 2006] A Goldsztejn. *A branch and prune algorithm for the approximation of nonlinear AE-solution sets*. In Proceedings of the ACM symposium on Applied Computing, pages 1650–1654, 2006. (Cited on page 45.)

[Goldsztejn 2007] A Goldsztejn and G Chabert. *A generalized interval LU decomposition for the solution of interval linear systems*. Numerical Methods and Applications, pages 312–319, 2007. (Cited on page 39.)

[Gordon 1993] N J Gordon, D J Salmond and A F M Smith. *Novel approach to nonlinear/non-Gaussian Bayesian state estimation*. In IEE Proceedings F on Radar and Signal Processing, volume 140, pages 107–113, April 1993. (Cited on page 18.)

[Gouzé 2000] J L Gouzé, A Rapaport and M Z Hadj-Sadok. *Interval observers for uncertain biological systems*. Ecological modelling, vol. 133, no. 1-2, pages 45–56, 2000. (Cited on page 23.)

[Grewal 2001] M Grewal and A Andrews. Kalman filtering: theory and practice using MATLAB. Wiley, 2001. (Cited on page 14.)

[Guitton 1906] E Guitton. *Démonstration de la formule* $\int_{-\infty}^{+\infty} exp(-x^2)dx = \sqrt{\pi}$. Nouvelles annales de mathématiques, vol. 6, pages 237–239, 1906. (Not cited.)

[Hable 2010] R Hable. *A Minimum Distance Estimator in an Imprecise Probability Model-Computational Aspects and Applications*. International Journal of Approximate Reasoning, vol. 51, no. 9, pages 1114–1128, November 2010. (Cited on page 9.)

[Hajiyev 1998] C M Hajiyev and F Caliskan. *Fault detection in flight control systems via innovation sequence of Kalman filter*. In UKACC International Conference on Control, pages 1528–1533, 1998. (Cited on page 101.)

[Hajiyev 1999] C M Hajiyev and F American Control Conference 1999 Proceedings of the 1999 Caliskan. *Fault detection in flight control systems based on the generalized*

variance of the Kalman filter innovation sequence. In Proceedings of the 1999 American Control Conference, volume 1, pages 109–113, 1999. (Cited on page 101.)

[Hall 2004] J W Hall. *Generation, combination and extension of random set approximations to coherent lower and upper probabilities*. Reliability Engineering and System Safety, vol. 85, no. 1-3, pages 89–101, July 2004. (Cited on page 9.)

[Hanebeck 2001] U D Hanebeck. *Recursive nonlinear set-theoretic estimation based on pseudo ellipsoids*. In International Conference on Multisensor Fusion and Integration for Intelligent Systems, pages 159–164, 2001. (Cited on page 22.)

[Hargreaves 2002] G Hargreaves. *Interval analysis in MATLAB*. Rapport technique, Manchester Centre for Computational Mathematics, 2002. (Cited on page 67.)

[He 1999] X F He and B Vik. *Use of extended interval Kalman filter on integrated GPS/INS system*. In Proceedings of the 12th International Technical Meeting of the Satellite Division of The Institute of Navigation, pages 1907–1914, 1999. (Cited on page 83.)

[Hero 1997] A Hero, Y Zhang and W Rogers. *Tomographic feature detection and classification using parallelotope bounded error estimation*. In IEEE International Conference on Acoustics, Speech, and Signal Processing, volume 4, pages 2849–2852, 1997. (Cited on page 10.)

[Hoehmann 2007] L Hoehmann and Kummert. *Laser range finder based obstacle tracking by means of a two-dimensional Kalman filter*. In International Workshop on Multidimensional (nD) Systems, pages 9–14, 2007. (Cited on page 14.)

[Hofbaur 2004] M W Hofbaur. Hybrid estimation of complex systems. Springer, 2004. (Cited on page 18.)

[Hou 1992] M Hou and P C Muller. *Design of observers for linear systems with unknown inputs*. IEEE Transactions on Automatic Control, vol. 37, no. 6, pages 871–875, June 1992. (Cited on page 15.)

[Hürzeler 1998] M Hürzeler and H R Künsch. *Monte Carlo approximations for general state-space models*. Journal of Computational and Graphical Statistics, vol. 7, no. 2, pages 175–193, 1998. (Cited on page 18.)

[Hutter 2003] F Hutter and R Dearden. *The gaussian particle filter for diagnosis of non-linear systems*. In Proceedings of the 14th International Conference on Principles of Diagnosis, pages 65–70, 2003. (Cited on page 18.)

[Ingimundarson 2009] A Ingimundarson, J M Bravo, V Puig, T Alamo and P Guerra. *Robust fault detection using zonotope-based set-membership consistency test*. In International Journal Of Adaptive Control And Signal Processing, pages 311–330. Univ Politecn Cataluna, Dept Engn Sistemes Automat & Informat Ind, Terrassa 08222, Spain, January 2009. (Cited on pages 1, 30 and 101.)

[Isermann 1984] R Isermann. *Process fault detection based on modeling and estimation methods. A survey.* Automatica, vol. 20, no. 4, pages 387–404, January 1984. (Cited on page 29.)

[Isermann 1997a] R Isermann. *Supervision, fault-detection and fault-diagnosis methods—an introduction*. Control Engineering Practice, vol. 5, no. 5, pages 639–652, 1997. (Cited on pages 25, 26 and 101.)

[Isermann 1997b] R Isermann and P Balle. *Trends in the application of model-based fault detection and diagnosis of technical Processes*. Control Engineering Practice, vol. 5, no. 5, pages 709–719, May 1997. (Cited on page 26.)

[Isermann 2005] R Isermann. *Model-based fault-detection and diagnosis–status and applications*. Annual Reviews in Control, vol. 29, pages 71–85, January 2005. (Cited on page 26.)

[Isermann 2006] R Isermann. Fault-diagnosis systems: An Introduction from Fault Detection to Fault Tolerance. Springer, 2006. (Cited on page 26.)

[Janssen 1999] M Janssen, Y Deville and P Hentenryck. *Multistep Filtering Operators for Ordinary Differential Equations*. In Principles and Practice of Constraint Programming, volume 1713, pages 246–260. Springer Berlin Heidelberg, Berlin, Heidelberg, 1999. (Cited on page 23.)

[Janssen 2003] M Janssen, P van Hentenryck and Y Deville. *A constraint satisfaction approach for enclosing solutions to parametric ordinary DIFFERENTIAL EQUATIONS*. SIAM Journal on Numerical Analysis, vol. 40, no. 5, pages 1896–1939, January 2003. (Cited on page 38.)

[Jauberthie 2009] C Jauberthie, R Pons, L Trave-Massuyes and P Goupil. *Detection of oscillatory failures by state estimation in a bounded-error context*. Rapport technique 09133, LAAS-CNRS, Toulouse, France, 2009. (Cited on page 33.)

[Jaulin 1993] L Jaulin and E Walter. *Set inversion via interval analysis for nonlinear bounded-error estimation*. Automatica, vol. 29, no. 4, pages 1053–1064, 1993. (Cited on pages 51, 53 and 88.)

[Jaulin 1994] L Jaulin. *Solution globale et garantie de problèmes ensemblistes; applications à l'estimation non linéaire et à la commande robuste*. PhD thesis, Université Paris XI Orsay, 1994. (Cited on page 21.)

[Jaulin 2001a] L Jaulin, I Braems, M Kieffer and E Walter. *Nonlinear state estimation using forward-backward propagation of intervals in an algorithm*. Scientific computing, validated numerics, interval methods, vol. 1, pages 191–204, 2001. (Cited on pages 1 and 39.)

[Jaulin 2001b] L Jaulin, M Kieffer, I Braems and E Walter. *Guaranteed non-linear estimation using constraint propagation on sets*. International Journal of Control, vol. 74, no. 18, pages 1772–1782, January 2001. (Cited on pages 55 and 89.)

[Jaulin 2001c] L Jaulin, M Kieffer, O Didrit and E Walter. Applied interval analysis. Springer-Verlag, January 2001. (Cited on page 48.)

[Jaulin 2001d] L Jaulin, M Kieffer, O Didrit and E Walter. Applied interval analysis: with examples in parameter and state estimation, robust control and robotics. An

emerging paradigm. Springer-Verlag, 1 édition, 2001. (Cited on pages 1, 2, 38, 41 and 85.)

[Jaulin 2006] L Jaulin. *Computing minimal-volume credible sets using interval analysis; application to bayesian estimation*. Signal Processing, IEEE Transactions on, vol. 54, no. 9, pages 3632–3636, 2006. (Cited on page 39.)

[Jaulin 2008] L Jaulin and G Chabert. *QUIMPER: un langage de programmation pour le calcul ensembliste; Application à l'automatique*. In 5ème Conférence Internationale Francophone d'Automatique, Bucarest, Roumanie, September 2008. (Cited on page 67.)

[Jaulin 2013] Luc Jaulin and Didier Henrion. *Contracting Optimally an Interval Matrix without Loosing Any Positive Semi-Definite Matrix Is a Tractable Problem*. Reliable Computing, vol. 11, no. 1, pages 1–17, May 2013. (Cited on page 85.)

[Jeffreys 1946] H Jeffreys. *An invariant form for the prior probability in estimation problems*. In Proceedings of the Royal Society of London, volume 186, pages 453–461, 1946. (Cited on page 6.)

[Julier 2004] S J Julier and J K Uhlmann. *Unscented Filtering and Nonlinear Estimation*. In Proceedings of the IEEE, volume 92, pages 401–422, March 2004. (Cited on pages 16 and 17.)

[Kai 2010] X Kai, CH L Wei and L D Liu. *Robust Extended Kalman Filtering for Nonlinear Systems With Stochastic Uncertainties*. IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, vol. 40, no. 2, pages 399–405, 2010. (Cited on pages 9 and 17.)

[Kalman 1960] R E Kalman. *A new approach to linear filtering and prediction problems*. Journal of Basic Engineering, no. 82, pages 35–45, 1960. (Cited on pages 1, 6, 12 and 73.)

[Kamali 2011] A J Kamali, L Longpre and M Koshelev. *Estimating mean under interval uncertainty and variance constraint*. In 2011 Annual Meeting of the North American on Fuzzy Information Processing Society (NAFIPS), pages 1–6, 2011. (Cited on page 38.)

[Karim 2008] J Karim, C Jauberthie and M Combacau. *Model-based fault detection method using interval analysis: Application to an aeronautic test bench*. In 19th International Workshop on Principles of Diagnosis DX, pages 269–273, 2008. (Cited on page 33.)

[Kearfott 1996] R B Kearfott and V Kreinovich. Applications of interval computations. Springer, 1996. (Cited on page 41.)

[Kesavan 2001] P Kesavan and J Lee. *A set based approach to detection and isolation of faults in multivariable systems*. Computers and Chemical Engineering, vol. 25, no. 7-8, pages 925–940, 2001. (Cited on page 103.)

[Kieffer 1999] M Kieffer, L Jaulin, E Walter and D Meizel. *Guaranteed mobile robot tracking using interval analysis*. In Proceedings of MISC'99: Workshop on Applications of

Interval Analysis to Systems and Control with Special Emphasis on Recent Advances in Modal Interval Analysis, pages 347–360, Girona, Spain, 1999. (Cited on page 1.)

[Klumpp 2009] V Klumpp and U D Hanebeck. *Bayesian estimation with uncertain parameters of probability density functions*. In 12th International Conference on Information Fusion, pages 1759–1766, 2009. (Cited on page 9.)

[Kosanam 2004] S Kosanam and D Simon. Kalman Filtering with Uncertain Noise Covariances. Cleveland State University, 2004. (Cited on page 10.)

[Koutsoukos 2003] X Koutsoukos, J Kurien and F Zhao. *Estimation of distributed hybrid systems using particle filtering methods*. Hybrid Systems: Computation and Control, pages 298–313, 2003. (Cited on page 18.)

[Kreinovich 2004] V Kreinovich. *Probabilities, Intervals, What Next? Optimization Problems Related to Extension of Interval Computations to Situations with Partial Information about Probabilities*. Journal of Global Optimization, vol. 29, no. 3, pages 265–280, 2004. (Cited on page 9.)

[Kreinovich 2007a] V Kreinovich, J BECK, C Ferregut, A Sanchez, G R Keller, M Averill and S A Starks. *Monte-Carlo-type techniques for processing interval uncertainty, and their potential engineering applications*. Reliable Computing, vol. 13, no. 1, pages 25–69, 2007. (Cited on page 20.)

[Kreinovich 2007b] V Kreinovich, L Longpre, S Starks, G Xiang, J BECK, R Kandathi, A NAYAK, S Ferson and J Hajacos. *Interval versions of statistical techniques with applications to environmental analysis, bioinformatics, and privacy in statistical databases*. Journal of Computational and Applied Mathematics, vol. 199, no. 2, pages 418–423, February 2007. (Cited on pages 38 and 39.)

[Kreinovich 2009] V Kreinovich, A Neumaier and G Xiang. *Towards a combination of interval and ellipsoid uncertainty*. Computational Technologies, vol. 13, no. 6, pages 5–16, 2009. (Cited on page 10.)

[Kreinovich 2011] V Kreinovich and H T Nguyen. *Interval-Valued Fuzzy Control in Space Exploration*. Bulletin for Studies and Exchanges on Fuzziness and its AppLications, no. 71, pages 55–64, July 2011. (Cited on page 38.)

[Kühn 1998] W Kühn. *Rigorously computed orbits of dynamical systems without the wrapping effect*. Computing, vol. 61, no. 1, pages 47–67, March 1998. (Cited on page 24.)

[Laaraiedh 2009] M Laaraiedh. *Implementation of Kalman Filter with Python Language*. The Python Papers, vol. 4, no. 2, pages 1834–3147, April 2009. (Cited on page 14.)

[Lalami 2006] A Lalami. *Generation of set membership tests for fault diagnosis and evaluation of their worst case sensitivit*. In Zhang Zhang, editeur, Fault Detection, Supervision and Safety of Technical Processes, pages 569–574. IFAC, Elsevier, August 2006. (Cited on page 28.)

[LeGland 2000] F LeGland and N Oudjane. *Stability and approximation of nonlinear filters in the Hilbert metric, and application to particle filters*. In Proceedings of the 39th IEEE Conference on Decision and Control, pages 1585–1590, 2000. (Cited on page 18.)

[Lesecq 2003] S Lesecq, A Barraud and K Dinh. *Numerical accurate computations for ellipsoidal state bounding*. In 11th Mediterranean Conference on Control and Automation, Rhodes, Greece, June 2003. (Cited on page 1.)

[Lhomme 1993] O Lhomme. *Consistency techniques for numeric CSPs*. In International Joint Conference on Artificial Intelligence, pages 232–238, 1993. (Cited on page 55.)

[Lhomme 1996] O Lhomme, R La Chantrerie, A Gotlieb and M Rueher. *Boosting the interval narrowing algorithm*. In Proceedings of the 1996 Joint International Conference and Symposium on Logic Programming, pages 378–392, Bonn, Germany, 1996. The MIT Press. (Cited on pages 59, 88 and 89.)

[Li 2001] P Li and V. Kadirkamanathan. *Particle filtering based likelihood ratio approach to fault diagnosis in nonlinear stochastic systems*. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, vol. 31, no. 3, pages 337–343, August 2001. (Cited on page 18.)

[Li 2008] B H Li, Ch Ch Li and J Si. *Interval recursive least-squares filtering with applications to video target tracking*. In Ivan Kadar, editeur, Proceedings of the SPIE: Signal Processing, Sensor Fusion, and Target Recognition XVII, volume 6968, pages 69681D–69681D–12, 2008. (Cited on pages 82 and 89.)

[Ljung 1999] L Ljung. System identification: theory for the user. Upper Saddle River, N.J., 2nd édition, January 1999. (Cited on page 29.)

[Lou 2003] K Y Lou and C H Goh. *Design and real-time implementation of a control system for port AGV application*. In IEEE International Symposium on Computational Intelligence in Robotics and Automation, volume 2, pages 527– 532, 2003. (Cited on page 14.)

[Luenberger 1964] D Luenberger. *Observing the State of a Linear System*. IEEE Transactions on Military Electronics, vol. 8, pages 74–80, January 1964. (Cited on pages 12 and 32.)

[Luenberger 1971] D Luenberger. *An introduction to observers*. IEEE Transactions on Automatic Control, vol. 16, no. 6, pages 596–602, 1971. (Cited on pages 12 and 15.)

[Macalak 2010] J Macalak. *Application of Kalman filtering to acoustic sound diagnostic measurement*. In MECHATRONIKA, 13th International Symposium, pages 84–86, 2010. (Cited on page 14.)

[Makkuni 1990] U K Makkuni. Residue Generation Techniques for Failure Diagnosis in Linear Systems. Master's thesis, Florida Atlantic University, 1990. (Cited on page 107.)

[Maksarov 1998] D Maksarov and Z Chalabi. *Computing bounds on greenhouse energy requirements using bounded error approach*. Control Engineering Practice, vol. 6, no. 8, pages 947–955, August 1998. (Cited on page 10.)

[Malti 2010] R Malti, T Raissi, M Thomassin and F Khemane. *Set membership parameter estimation of fractional models based on bounded frequency domain data*. Commun Nonlinear Sci Numer Simulat, vol. 15, no. 4, pages 927–938, January 2010. (Cited on pages 22 and 30.)

[Markov 1996] S Markov and E D Popova. *On the solution of linear algebraic equations involving interval coefficients*. Iterative Methods In Linear Algebra, II, Imacs Series In Computational And Applied Mathematics, pages 216–225, 1996. (Cited on page 39.)

[Marx 2010] B Marx, D Maquin and J. Ragot. *State estimation and fault detection of uncertain systems based on an interval approach*. In Conference on Control and Fault-Tolerant Systems (SysTol), pages 720–725. IEEE, 2010. (Cited on pages 33 and 107.)

[Masreliez 1977] C Masreliez and R Martin. *Robust bayesian estimation for the linear model and robustifying the Kalman filter*. IEEE Transactions on Automatic Control, vol. 22, no. 3, pages 361–371, 1977. (Cited on page 13.)

[Mcilraith 2000] S A Mcilraith. *Diagnosing hybrid systems: A bayesian model selection approach*. In Proceedings of the 11th International Workshop on Principles of Diagnosis, pages 140–146, 2000. (Cited on page 18.)

[Melquiond 2008] G Melquiond. *Proving Bounds on Real-Valued Functions with Computations*. Automated Reasoning, vol. 5195, pages 2–17, 2008. (Cited on page 38.)

[Milanese 1982] M Milanese and G Belforte. *Estimation theory and uncertainty intervals evaluation in presence of unknown but bounded errors: Linear families of models and estimators*. IEEE Transactions on Automatic Control, vol. 27, no. 2, pages 408–414, 1982. (Cited on page 10.)

[Milanese 1991] M Milanese and A Vicino. *Optimal estimation theory for dynamic systems with set membership uncertainty*. Automatica, vol. 27, no. 6, pages 997–1009, November 1991. (Cited on page 10.)

[Milanese 2004] M Milanese and C Novara. *Nonlinear Set Membership prediction of river flow*. Systems and Control Letters, vol. 53, no. 1, pages 31–39, 2004. (Cited on page 1.)

[Moore 1966] R E Moore. Interval analysis. Prentice-Hall, Englewood Cliffs, January 1966. (Cited on pages 1, 10, 23, 38, 41 and 43.)

[Moore 1979] R E Moore and F Bierbaum. Methods and Applications of Interval Analysis. SIAM Studies in Applied Mathematics 2, 1979. (Cited on page 50.)

[Morrison 1991] I J Morrison and P J W Rayner. *The application of Volterra series to signal estimation*. In International Conference on Acoustics, Speech, and Signal Processing, volume 2, pages 1481–1484, April 1991. (Cited on page 18.)

[Mosallaei 2008] M Mosallaei and K Salahshoor. *Sensor Fault Detection using Adaptive Modified Extended Kalman Filter Based on Data Fusion Technique*. In 4th International Conference on Information and Automation for Sustainability, pages 513–518, 2008. (Cited on page 101.)

[Muenchhof 2009] M Muenchhof, M Beck and R Isermann. *Fault-tolerant actuators and drives—Structures, fault detection principles and applications*. Annual Reviews in Control, vol. 33, no. 2, pages 136–148, January 2009. (Cited on page 26.)

[Nandam 1990] P K Nandam and P C Sen. *A comparative study of a Luenberger observer and adaptive observer-based variable structure speed control system using a self-controlled synchronous motor*. IEEE Transactions on Industrial Electronics, vol. 37, no. 2, pages 127–132, 1990. (Cited on page 15.)

[Nedialkov 1999a] N S Nedialkov. Computing Rigorous Bounds on the Solution of an Initial Value Problem for an Ordinary Differential Equation [microform]. National Library of Canada, 1999. (Cited on page 23.)

[Nedialkov 1999b] N S Nedialkov. *Validated solutions of initial value problems for ordinary differential equations*. Applied Mathematics and Computation, vol. 105, no. 1, pages 21–68, October 1999. (Cited on page 23.)

[Neumaier 2003] A Neumaier. *Taylor forms - use and limits*. Reliable Computing, pages 9–43, 2003. (Cited on page 49.)

[Noack 2008] B Noack, V Klumpp and D Brunn. *Nonlinear Bayesian estimation with convex sets of probability densities*. In 11th International Conference on Information Fusion, pages 1–8, 2008. (Cited on page 9.)

[Noack 2009] B Noack, V Klumpp and U D Hanebeck. *State estimation with sets of densities considering stochastic and systematic errors*. In 12th International Conference on Information Fusion, pages 1751–1758, 2009. (Cited on page 9.)

[Olive 2010] X Olive. *FDI(R) for satellite at Thales Alenia Space how to deal with high availability and robustness in space domain?* In Conference on Control and Fault-Tolerant Systems (SysTol), pages 837–842. IEEE, 2010. (Cited on page 114.)

[Oppenheim 2008] G Oppenheim and A Philippe. *The particle filters and their applications*. Chemometrics and Intelligent Laboratory Systems, vol. 91, no. 1, pages 87–93, March 2008. (Cited on page 18.)

[Orlowska-Kowalska 1989] T Orlowska-Kowalska. *Application of extended Luenberger observer for flux and rotor time-constant estimation in induction motor drives*. In IEE Proceedings D on Control Theory and Applications, pages 324–330, 1989. (Cited on page 15.)

[Paşca 2010] I Paşca. *Formally verified conditions for regularity of interval matrices*. Intelligent Computer Mathematics, pages 219–233, 2010. (Cited on page 65.)

[Petersen 1999] I R Petersen and A V Savkin. Robust Kalman Filtering For Signals and Systems with Large Uncertainties. Springer, 1999. (Cited on page 24.)

[Ploix 1999] S Ploix, O Adrot and J Ragot. *Parameter uncertainty computation in static linear models*. In Proceedings of the 38th IEEE CDC, pages 1916–1921, Phoenix, Arizona, USA, 1999. (Cited on pages 22 and 29.)

[Ploix 2001] S Ploix, O Adrot and J Ragot. *Bounding approach to the diagnosis of uncertain static systems*. In 4th Symposium on Fault Detection, Supervision and Safety for Technical Processes, Safeprocess'2000, volume 1, pages 151–156, June 2001. (Cited on page 28.)

[Ploix 2006] S Ploix and O Adrot. *Parity relations for linear uncertain dynamic systems*. Automatica, vol. 42, no. 9, pages 1553–1562, January 2006. (Cited on page 31.)

[Poignet 2003a] P Poignet and N Ramdani. *Robust Estimation of Parallel Robot with Interval Analysis*. In 42nd IEEE Conference on Decision and Control, Maui, Hawaii (USA), 2003. (Cited on page 38.)

[Poignet 2003b] P Poignet, N Ramdani and O A Vivas. *Ellipsoidal estimation of parallel robot dynamic parameters*. In Proceedings of International Conference on Intelligent Robots and Systems, volume 3, pages 3300– 3305, 2003. (Cited on page 10.)

[Pons 2008] R Pons, C Jauberthie, L Trave-Massuyes and P Goupil. *Interval analysis based learning for fault model identification. Application to control surfaces oscillatory failures*. In 19th International Workshop on Principles of Diagnosis, DX'08, 2008. (Cited on page 30.)

[Popova 1998] E D Popova. *Algebraic Solutions to a Class of Interval Equations*. Journal of Universal Computer Science, vol. 4, no. 1, pages 48–67, 1998. (Cited on page 45.)

[Price 1982] M G Price and G Cook. *Identification/Observation Using an Extended Luenberger Observer*. IEEE Transactions on Industrial Electronics, no. 4, pages 279–287, 1982. (Cited on page 15.)

[Puig 2003] V Puig, J Saludes and J Quevedo. *Worst-case simulation of discrete linear time-invariant interval dynamic systems*. Reliable Computing, vol. 9, no. 4, pages 251–290, August 2003. (Cited on pages 11, 22 and 28.)

[Ra 2008] W S Ra and I H Whang. *Stochastic Robust Kalman Filtering for Linear Time-Varying Systems with a Multiplicative Measurement Noise*. In Proceedings of the 17th World Congress of the International Federation of Automatic Control, volume 17, pages 12546–12551, COEX, Korea, South, 2008. (Cited on page 24.)

[Raissi 2004] T Raissi, N Ramdani and Y Candau. *Set membership state and parameter estimation for systems described by nonlinear differential equations*. Automatica, vol. 40, pages 1771–1777, January 2004. (Cited on pages 22, 28 and 33.)

[Ramdani 2005a] N Ramdani. *Méthodes ensemblistes pour l'estimation*. Habilitation à diriger des recherches, 2005. (Cited on page 22.)

[Ramdani 2005b] N Ramdani and P Poignet. *Robust Dynamic Experimental Identification of Robots With Set Membership Uncertainty*. IEEE/ASME Transactions on Mechatronics, vol. 10, no. 2, pages 253–256, April 2005. (Cited on page 38.)

[Ramdani 2005c] N Ramdani, T Raïssi, Y Candau and L Ibos. *Complex Interval Constraint Propagation for Non Linear Bounded-Error Parameter Identification*. In Proceedings of the 44th IEEE Conference of Decision and Control, pages 3590–3595, 2005. (Cited on page 22.)

[Ramdani 2010] N Ramdani. *Computing reachable sets for uncertain nonlinear hybrid systems using interval constraint propagation techniques*. Nonlinear Analysis: Hybrid Systems, pages 1–23, June 2010. (Cited on pages 49 and 89.)

[Rami 2008] M A Rami, C H Cheng and C de Prada. *Tight Robust interval observers: an LP approach*. In 47th IEEE Conference on Decision and Control, pages 2967–2972, 2008. (Cited on pages 22 and 33.)

[Rao 1976] C Rao. *Estimation of parameters in a linear model*. The Annals of Statistics, vol. 4, no. 6, pages 1023–1037, 1976. (Cited on page 21.)

[Rapaport 2003] A Rapaport and J L Gouzé. *Parallelotopic and practical observers for nonlinear uncertain systems*. International Journal of Control, vol. 76, no. 3, pages 237–251, January 2003. (Cited on page 23.)

[Ratschek 1985] H Ratschek. *Inclusion functions and global optimization*. Mathematical Programming, vol. 33, no. 3, pages 300–317, 1985. (Cited on page 48.)

[Ratz 1995] D Ratz and T Csendes. *On the selection of subdivision directions in interval branch-and-bound methods for global optimization*. Journal of Global Optimization, vol. 7, no. 2, pages 183–207, 1995. (Cited on page 53.)

[Rauh 2009] A Rauh, J Minisini and E P Hofer. *Verification Techniques for Sensitivity Analysis and Design of Controllers for Nonlinear Dynamic Systems with Uncertainties*. International Journal of Applied Mathematics and Computer Science, vol. 19, no. 3, pages 425–439, September 2009. (Cited on page 38.)

[Reichle 2002] R Reichle, J Walker and R Koster. *Extended versus ensemble Kalman filtering for land data assimilation*. Journal of Hydrometeorology, vol. 3, no. 5, pages 728–741, 2002. (Cited on page 17.)

[Ribot 2007] P Ribot, C Jauberthie and L Trave-Massuyes. *State estimation by interval analysis for a nonlinear differential aerospace model*. In Proceedings of European Control Conference, pages 4839–4844, Kos, Greece, 2007. (Cited on page 1.)

[Rico 2010] A Rico. *Imprecise expectations for imprecise linear filtering*. International Journal of Approximate Reasoning, vol. 51, no. 8, pages 933–947, 2010. (Cited on page 9.)

[Rienmüller 2013] T Rienmüller, M W Hofbaur and L Trave-Massuyes. *Mode set focused hybrid estimation*. International Journal of Applied Mathematics and Computer Science, vol. 23, no. 1, pages 131–144, 2013. (Cited on page 115.)

[Rihm 1994] R Rihm. *Interval methods for initial value problems in ODEs*. Topics in Validated Computations, pages 173–207, 1994. (Cited on page 23.)

[Rohn 1993] J Rohn. *Inverse interval matrix*. SIAM Journal on Numerical Analysis, vol. 30, no. 3, pages 864–870, 1993. (Cited on pages 65, 66 and 88.)

[Rohn 2009] J Rohn. *Forty necessary and sufficient conditions for regularity of interval matrices: A survey*. Electronic Journal of Linear Algebra, vol. 18, pages 500–512, 2009. (Cited on page 65.)

[Rump 1999] S M Rump. *INTLAB - INTerval LABoratory*. In Tibor Csendes, editeur, Developments in Reliable Computing, pages 77–104. Kluwer Academic Publishers, Dordrecht, 1999. (Cited on pages 65, 67 and 94.)

[Rump 2006] S M Rump. *Verification of positive definiteness*. BIT Numerical Mathematics, vol. 46, no. 2, pages 433–452, 2006. (Cited on page 65.)

[Shamma 1999] J Shamma and K Tu. *Set-valued observers and optimal disturbance rejection*. IEEE Transactions on Automatic Control, vol. 44, no. 2, pages 253–264, 1999. (Cited on page 22.)

[Shevtshenko 2009] E Shevtshenko. *Decision support under uncertainties based on robust Bayesian networks in reverse logistics management*. International Journal of Computer Applications in Technology, vol. 36, pages 247–258, 2009. (Cited on page 39.)

[Shi 1999] P Shi, E Boukas and R Agarwal. *Kalman filtering for continuous-time uncertain systems with Markovian jumping parameters*. IEEE Transactions on Automatic Control, vol. 44, no. 8, pages 1592–1597, 1999. (Cited on page 24.)

[Shi 2007] L Shi, K H Johansson and R M Murray. *Kalman filtering with uncertain process and measurement noise covariances with application to state estimation in sensor networks*. In IEEE Conference on Control Applications, pages 1031–1036, 2007. (Cited on page 10.)

[Simani 2003] S Simani, C Fantuzzi and R Patton. Model-based fault diagnosis in dynamic systems using identification techniques. Springer, January 2003. (Cited on page 29.)

[Souto 2008] R F Souto and J Y Ishihara. *Enhanced robust Kalman predictor for discrete-time systems with uncertain correlated noises*. In Proceedings of the 47th IEEE Conference of Decision and Control, pages 1660–1665, 2008. (Cited on page 24.)

[Stancu 2005] A Stancu, V Puig, P Cugueró and J Quevedo. *Benchmarking on Approaches to Interval Observation Applied to Robust Fault Detection*. In Global Optimization and Constraint Satisfaction, pages 171–191. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005. (Cited on pages 28 and 32.)

[Staroswiecki 2001] M Staroswiecki and G Comtet-Varga. *Analytical redundancy relations for fault detection and isolation in algebraic dynamic systems*. Automatica, vol. 37, pages 687–699, 2001. (Cited on page 26.)

[Storvik 2002] G Storvik. *Particle filters for state-space models with the presence of unknown static parameters*. IEEE Transactions on Signal Processing, vol. 50, no. 2, pages 281–289, 2002. (Cited on page 19.)

[Trave-Massuyes 2001] L Trave-Massuyes, T Escobet, R Pons and S Tornil. *The Ca-En diagnosis system and its automatic modelling method*. Computacion y Sistemas Journal, vol. 5, no. 2, pages 128–143, 2001. (Cited on pages 107 and 120.)

[Tudoroiu 2011] N Tudoroiu and M Zaheeruddin. *Joint UKF estimator for fault detection and isolation in Heating Ventilation and Air Conditioning systems*. In 5th International Symposium on Computational Intelligence and Intelligent Informatics (ISCIII), pages 53–58, 2011. (Cited on page 17.)

[Utkin 2009] L Utkin. *Computing expectations with continuous p-boxes: Univariate case*. International Journal of Approximate Reasoning, vol. 50, no. 5, pages 778–798, 2009. (Cited on page 9.)

[Uwamusi 2008] S E Uwamusi. *On the interval hull of solution sets of parametrised nonlinear equations*. Scientific Research and Essays, vol. 3, no. 9, pages 383–389, 2008. (Cited on page 51.)

[Verma 2003] V Verma and S Thrun. *Variable resolution particle filter*. In Proceedings of International Joint Conference on Artificial Intelligence, August 2003. (Cited on page 18.)

[Verma 2004] V Verma, G Gordon and R Simmons. *Particle filters for rover fault diagnosis*. IEEE Robotics and Automation Magazine Special Issue on Human Centered Robotics and Dependability, 2004. (Cited on page 18.)

[Walley 1991] P Walley. Statistical reasoning with imprecise probabilities. Chapman and Hall, 1991. (Cited on pages 9 and 84.)

[Walley 2000] P Walley. *Towards a unified theory of imprecise probability*. International Journal of Approximate Reasoning, vol. 24, no. 2-3, pages 125–148, May 2000. (Cited on page 9.)

[Walster 2004] G W Walster and E R Hansen. Global Optimization Using Interval Analysis. CRC Press, 2004. (Cited on page 54.)

[Walter 1990] E Walter and H Piet-Lahanier. *Estimation of parameter bounds from bounded-error data: a survey*. Mathematics and Computers in Simulation, vol. 32, no. 5-6, pages 449–468, January 1990. (Cited on pages 21 and 29.)

[Walter 1997a] E Walter and L Pronzato. Identification de Modèles Paramétriques à Partir de Données Expérimentales. Masson, 1997. (Cited on pages 21 and 22.)

[Walter 1997b] E Walter and L Pronzato. Identification of parametric models. Springer, 1997. (Cited on page 29.)

[Wang 1997] H Wang, Z Huang and S Daley. *On the use of adaptive updating rules for actuator and sensor fault diagnosis*. Automatica, vol. 33, no. 2, pages 217–225, January 1997. (Cited on page 32.)

[Wang 2007] Y Wang and B Nnaji. *Solving interval constraints by linearization in computer-aided design*. Reliable Computing, vol. 13, no. 2, pages 211–244, 2007. (Cited on page 39.)

[Wang 2008a] Y Wang. *Imprecise probabilities with a generalized interval form*. In Third International Workshop on Reliable Engineering Computing (REC), NSF Workshop on Imprecise Probability in Engineering Analysis and Design, 2008. (Cited on page 9.)

[Wang 2008b] Y Wang. *Interpretable Interval Constraint Solvers in Semantic Tolerance Analysis*. Computer-Aided Design and Applications, vol. 5, no. 5, page 654, 2008. (Cited on page 38.)

[Welch 2001] G Welch and G Bishop. *An Introduction to the Kalman Filter*. Rapport technique TR 95-041, University of North Carolina at Chapel Hill, Department of Computer Science, Los Angeles, California USA, 2001. (Cited on pages 12 and 17.)

[Wells 1995] C Wells. The Kalman Filter in Finance. Springer, November 1995. (Cited on page 14.)

[Willsky 1976] A Willsky. *A survey of design methods for failure detection in dynamic systems*. Automatica, vol. 12, no. 4, pages 601–611, January 1976. (Cited on page 26.)

[Xie 1994] L H Xie, Y C Soh and C E de Souza. *Robust Kalman filtering for uncertain discrete-time systems*. IEEE Transactions on Automatic Control, vol. 39, no. 6, pages 1310–1314, 1994. (Cited on page 24.)

[Xu 2005] J W Xu, D Erdogmus and J C Principe. *Minimum error entropy Luenberger observer*. In Proceedings of 2005 American Control Conference, pages 1923–1928, 2005. (Cited on page 15.)

[Xu 2009] H Xu and S Mannor. *A Kalman Filter Design Based on the Performance/Robustness Tradeoff*. IEEE Transactions on Automatic Control, vol. 54, no. 5, page 1171, 2009. (Cited on page 24.)

[Xue 2007] W Xue, Y Q Guo and Zhang X D. *A Bank of Kalman Filters and a Robust Kalman Filter Applied in Fault Diagnosis of Aircraft Engine Sensor/Actuator*. In Second International Conference on Innovative Computing, Information and Control, page 10, 2007. (Cited on pages 17 and 103.)

[Yager 1999] R Yager and V Kreinovich. *Decision making under interval probabilities*. International Journal of Approximate Reasoning, page 848, 1999. (Cited on page 9.)

[Yang 2009] F W Yang and Y M Li. *Set-membership filtering for systems with sensor saturation*. Automatica, vol. 45, no. 8, pages 1896–1902, June 2009. (Cited on page 22.)

[Zeitz 1987] M Zeitz. *The extended Luenberger observer for nonlinear systems*. Systems and Control Letters, vol. 9, no. 2, pages 149–156, August 1987. (Cited on page 15.)

[Zhang 1995] X F Zhang, A W Heemink and J C H van Eijkeren. *Performance robustness analysis of Kalman filter for linear discrete-time systems under plant and noise uncertainty*. International Journal of Systems Science, vol. 26, no. 2, pages 257–275, February 1995. (Cited on page 93.)

[Zhu 1999] Y Zhu. *Efficient recursive state estimator for dynamic systems without knowledge of noise covariances*. IEEE Transactions on Aerospace and Electronic Systems, vol. 35, no. 1, pages 102–114, January 1999. (Cited on page 14.)

[Zhu 2000] X Zhu, Y C Soh and L H Xie. *Robust Kalman filter design*. In Proceedings of the 39th IEEE Conference on Decision and Control, 2000, pages 3813–3818, 2000. (Cited on page 24.)

[Zhu 2006] Y Zhu and B Li. *Optimal interval estimation fusion based on sensor interval estimates with confidence degrees*. Automatica, vol. 42, no. 1, pages 101–108, 2006. (Cited on page 10.)

[Zhuk 2007] S Zhuk. *Recursive state estimation for noncausal discrete-time descriptor systems under uncertainties*. Optimization and Control, November 2007. (Cited on page 22.)

[Zhuk 2008] S Zhuk. *Set-membership state estimation framework for uncertain linear differential-algebraic equations*. In Conference on Differential and Difference Equations and Applications: Differential equations and Topology, Moscow, October 2008. (Cited on page 22.)

[Ziegler 1995] G M Ziegler. Lectures on Polytopes. Springer, 1995. (Cited on page 10.)


incertitudes
algorithme
filtre
iIKF
propagation
inversion
contraintes
intervalles
mesure
IKF
bruits
matrices
mécanismes
modélisation
Kalman
structure
détection
possibles
systèmes
mis
récursive
problème
noté
également
défauts
limiter
requiert
différents
dynamiques
perturbateurs
tolérances
erreurs
autre
efficacité
exemple
inévitable
optimaux
basés
liées
plusieurs
récentes
solutions
problèmes
phénomène
méconnaissance
effet
bornées
spécification
pessimisme
certaines
analyse
exemples
linéaires
originale
covariance
basé
amélioriations
techniques
Chen
prise
corrompues
discret
ailleurs
classique
basées
résultat
prêtent
via
autres
thèse
SIVIA
estimés
motivés
détecté
intégration
observations
précédent
vaut
significatives
mesures
obtenir
intervalle
paramètres
Set
démontré
proposons
Cet
évite
couple
caractérisent
semi-fermée
permet
garantie
Intervalle
bornes
proposé
temps
Partant
jeu
sans
méthode
ensembliste
existence
Analysis
Interval
phénomènes
type
permettent
avantages
contrôlant
ci-dessus
Inversion
valeur
travers
œuvre
tous
statistiques
certains
contrairement
due
développé
quant
pa
vieillissement
oeuvre
boucle
dès
mieux
défaut
ensemble
compte
dynamique
celui-ci
délivre
réalimenter
stratégie
mettant
statistique
surestimation
garanti
Alors
attribut
tout exposés
encadrement
comme
travail
traitons
bien
dynamiques
tolérances